DOCUMENT RESUME

ED 081 619 SE 016 700

TITLE	M.E.T.R.O.-Apex Gaming Simulation, Volume 28 (OS/360 Version).
INSTITUTION	Michigan Univ., Ann Arbor. Environmental Simulation Lab.; University of Southern California, Los Angeles. COMEX Research Project.
PUB DATE	[73]
NOTE	130p.
EDRS PRICE	MF-$0.65 HC-$6.58
DESCRIPTORS	*Air Pollution Control; City Problems; *Computer Programs; *Environmental Education; Instructional Materials; Management Games; *Manuals; Models; Pollution; *Simulation; Urban Environment
IDENTIFIERS	METRO APEX

ABSTRACT

Operator's instructions and technical support materials needed for processing the M.E.T.R.O.-APEX (Air Pollution Exercise) game decisions on an IBM 360 computer are compiled in this volume. M.E.T.R.O.-APEX is a computerized college and professional level "real world" simulation of a community with urban and rural problems, industrial activities, and air pollution difficulties. This manual takes the game operator from the worksheets filled out by players of the game through the complete card deck which must be submitted for computer input. In addition, it discusses the various program options and utility support features of the M.E.T.R.O.-APEX system. The computer program is in Fortran IV. A related document is ED 064 531 which presents the same computer operator's manual for use on an IBM 1130-2B computer. (BL)

ED 081619

U S DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

M.E.T.R.O.-APEX GAMING SIMULATION

VOLUME 28
(OS/360 Version)

Table of Contents

Chapter 1. Introduction.....1

Chapter 2. Preparing the Computer Input.....2

Chapter 3. Additional Game Director Input.....39

Chapter 4. Error Messages.....47

Chapter 5. The FILEPRNT Utility Program & Other Options.64

Chapter 6. METRO-APEX Data File Organization & Description.....75

FILMED FROM BEST AVAILABLE COPY

SE 016 700

M.E.T.R.O.-APEX GAME SIMULATION

Volume 1: Game Director's Manual
Volume 2: Computer Operator's Manual
Volume 3: Pre-Game Handouts
Volume 4: Air Pollution Control Officer's Manual
Volume 5: City Politician's Manual
Volume 6: County Politician's Manual
Volume 7: Industrialist's Manual: #1, Shear Power Company
Volume 8: Industrialist's Manual: #2, People's Pulp Plant
Volume 9: Industrialist's Manual: #3, Rusty's Iron Foundry
Volume 11: Industrialist's Manual: #5, Caesar's Rendering Plant
Volume 12: Industrialist's Manual: #6, Dusty Rhodes Cement Company
Volume 14: Developer's Manual: #1
Volume 15: Developer's Manual: #2
Volume 16: Developer's Manual: #3
Volume 17: Developer's Manual: #4
Volume 18: Developer's Manual: #5
Volume 19: Developer's Manual: #6
Volume 20: Developer's Manual: #7
Volume 21: City Planner's Manual
Volume 22: County Planner's Manual
Volume 23: Reference Materials
Volume 24: Legal References: Air Pollution Control Legislation
Volume 25: Photo Atlas
Volume 26: Forms: Worksheets
Volume 27: Forms: Coding Sheets

PREFACE

The M.E.T.R.O.-APEX project anticipated by several years the national concern for environmental problems which are now so much in vogue.

The emphasis in M.E.T.R.O.-APEX is on air pollution (or more properly, training in air quality management). The context for this training is a metropolitan region designed to insure a more valid perspective for the technical training concurrently supplied than that normally achieved through conventional teaching techniques.

The substance and nature of M.E.T.R.O.-APEX have been specifically designed for this task. Inspiration, and much technical input have come from a similar exercise (M.E.T.R.O.) developed somewhat earlier at The University of Michigan.

The current version, M.E.T.R.O.-APEX, is the result of close and continuing cooperation between the COMEX Research Project, School of Public Administration, The University of Southern California and the Environmental Simulation Laboratory, School of Natural Resources, The University of Michigan. The staff of both organizations have worked very closely throughout the project and share equally in any accolades or trauma that might result.

One central task remains for the successful completion of the project. The many potential users across the nation, and internationally (it has already been translated into French), will need special assistance in operationalizing the gaming-simulation on their home turf. Every effort has been made to make the exercise straightforward to operate. Nevertheless it is remarkably complex and special skills must be acquired for the game operators, the computer operators, and the role advisors; this is necessary with regard to both the general philosophy of use and the technical areas.

The two institutions intend to provide assistance to potential users to permit and encourage proper and frequent use of the exercise.

Dr. Richard D. Duke, Director
Environmental Simulation Laboratory
School of Natural Resources
The University of Michigan
Ann Arbor, Michigan

M.E.T.R.O.-APEX STAFF

Environmental Simulation Laboratory
School of Natural Resources
The University of Michigan

Dr. Richard D. Duke,
Principal Investigator

Roy I. Miller,
Project Director

Stewart D. Marquis, Research Design
Donald F. Kiel, Systems Analyst

Staff

Thomas Borton

Anne E. Cochran

Ferdinand Dijkstra

Katherine Fenn

David Kaueper

Marilyn Miller

Ellen Pechman

James Reeds

David Specht

Comex Research Project
School of Public Administration
The University of Southern California

Dr. K. William Leffland,
Principal Investigator

Richard K. McGinty,
Project Director

Alan Kreditor, Research Design
Ira Robinson, Research Design

Staff

Aubrey Boyd

Jolene Elliott

Corinne Floyd

Alan Forrest

Mark James

Frank King

Paula Magzamin

Charles Pratt

Gilbert Siegel

Chapter 1

INTRODUCTION

This volume of the operator's kit contains the instructions and technical support material needed for processing the M.E.T.R.O.-APEX game decisions on an IBM 360 computer.

The 360 series of IBM computers includes a wide range of sizes, speeds and variety of peripheral equipment (card readers, printers, tape drives, disk storage units and other off-line storage devices). Despite this complexity, no special skills in operating a computer or in computer programming are required to prepare M.E.T.R.O.-APEX player decisions for submission to the computer. Because of variations in Operating System control cards and in system configuration from one installation to another, the basic sections of this manual are written on the assumption that the M.E.T.R.O.-APEX programs have been installed and are correctly operating at your computer installation. Where certain instructions may be installation dependent, space has been provided for you to write in the correct procedure for your installation after consultation with a representative from that installation.

This volume takes the game operator from the worksheets filled out by players of the game through the complete card deck which must be submitted for computer input. It discusses the various program options and utility support features of the M.E.T.R.O.-APEX system.

Chapter 2

PREPARING THE COMPUTER INPUT

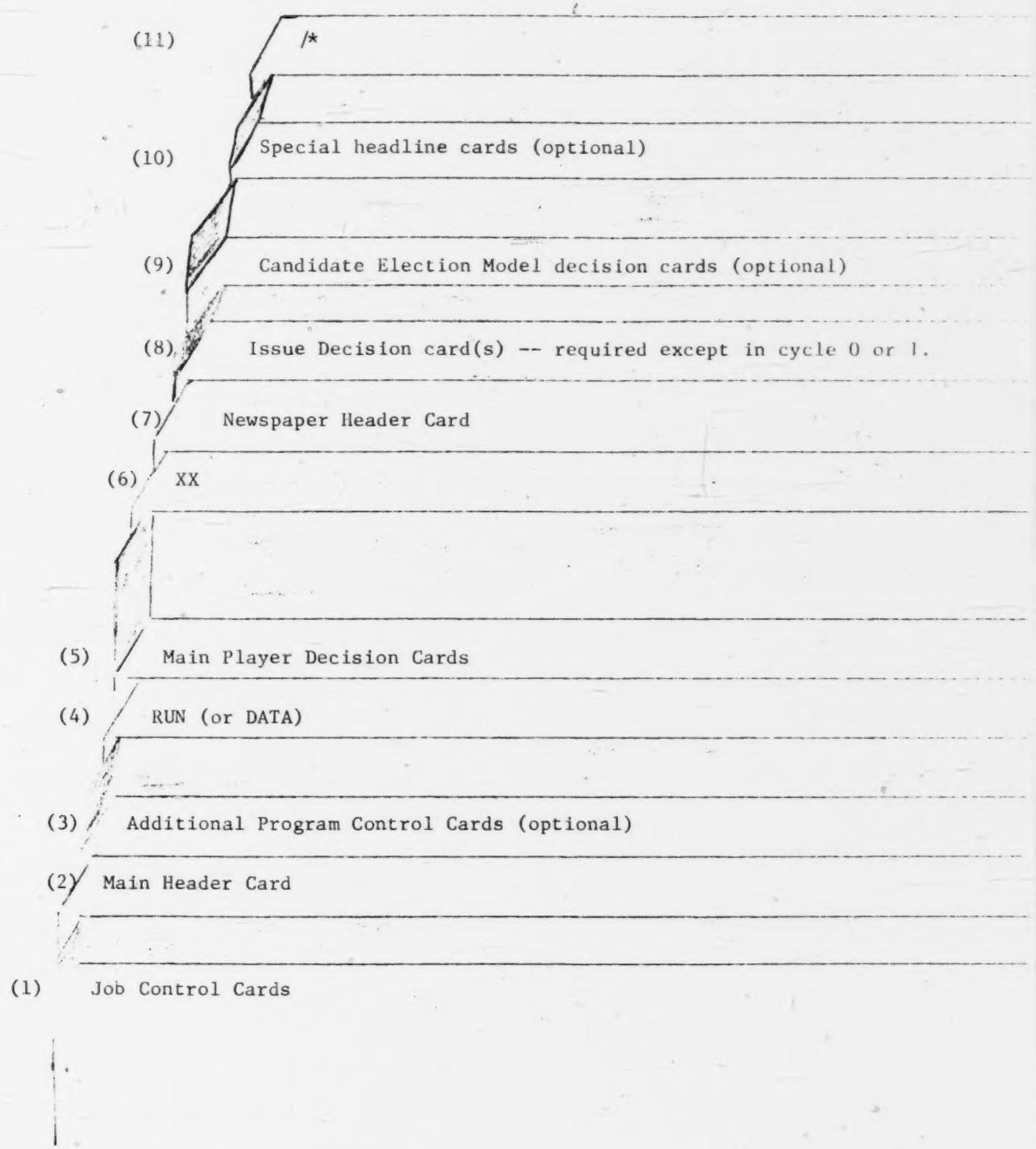
The first stop in the computer processing is the preparation of the input cards. There are eleven possible sets of input cards. Figure 1 shows the final arrangement of these sets of cards in the order in which they should be arranged for submission of a complete run to the computer. Some of these sets of cards may be only a single card, but they are shown separately because they must be included in exactly the location shown. Not all of the eleven sets of cards will always be required for the program to operate correctly; the following discussion will indicate the circumstances in which they may be omitted.

There are two main sections of the M.E.T.R.O.-APEX cycle processing program. The first section processes the main player decisions and produces the bulk of the computer output. The second section prints the M.E.T.R.O.-APEX NEWS, the newspaper based on factors calculated in the first section and on actions taken by players on issues raised in the newspaper the preceding cycle. The second section also includes the Candidate Election Model.

Since the main player decision cards (input set #5) constitute the bulk of the program input, are the most time-consuming to prepare and the most frequent sources of error, they should be prepared first. They may then be submitted for running (along with input sets 1 through 6) independently of the newspaper and issue input sets (7 through 11). The program will check the player decision cards for many types of inconsistencies and other incorrect coding and, unless you instruct otherwise, will terminate processing if any errors are found. You may even instruct the program to stop after reading the main data cards even if no program detectable errors are found so that you may double-check the list of the cards printed by the computer.

Figure 1

M.E.T.R.O.-APEX JOB INPUT DECK


(11) /*
(10) Special headline cards (optional)
(9) Candidate Election Model decision cards (optional)
(8) Issue Decision card(s) -- required except in cycle 0 or 1.
(7) Newspaper Header Card
(6) XX
(5) Main Player Decision Cards
(4) RUN (or DATA)
(3) Additional Program Control Cards (optional)
(2) Main Header Card
(1) Job Control Cards

A skilled operator can prepare the input sets for the newspaper and issue section of the program (sets 7 through 11) while the first section of the program is running on the computer and the newspaper can be printed in a separate job, bypassing the first main section of the program. Figure 2 shows the basic flow of the program and how these two sections of the program interact.

The decision as to whether you wish to prepare and submit the complete program input deck (1-11) or submit the data in two sections will usually depend on the type of game run you are conducting. If you are playing only one cycle a day (or less frequently) you may prefer to prepare all of the input cards and assemble the complete data deck for submission. This is recommended particularly for the first few times you run the game, since it involves fewer changes in the program option cards, less card shuffling and confusion and fewer chances of error in assembling the entire input deck.

However, if you are trying to play two or more cycles in a day and attempting to minimize the time between cycles, the two step procedure will probably be required to reduce turnaround time between submission of your job to the computer and getting back the results.

The rest of this chapter will discuss the input data card sets in detail in the order in which they are shown in Figure 1.

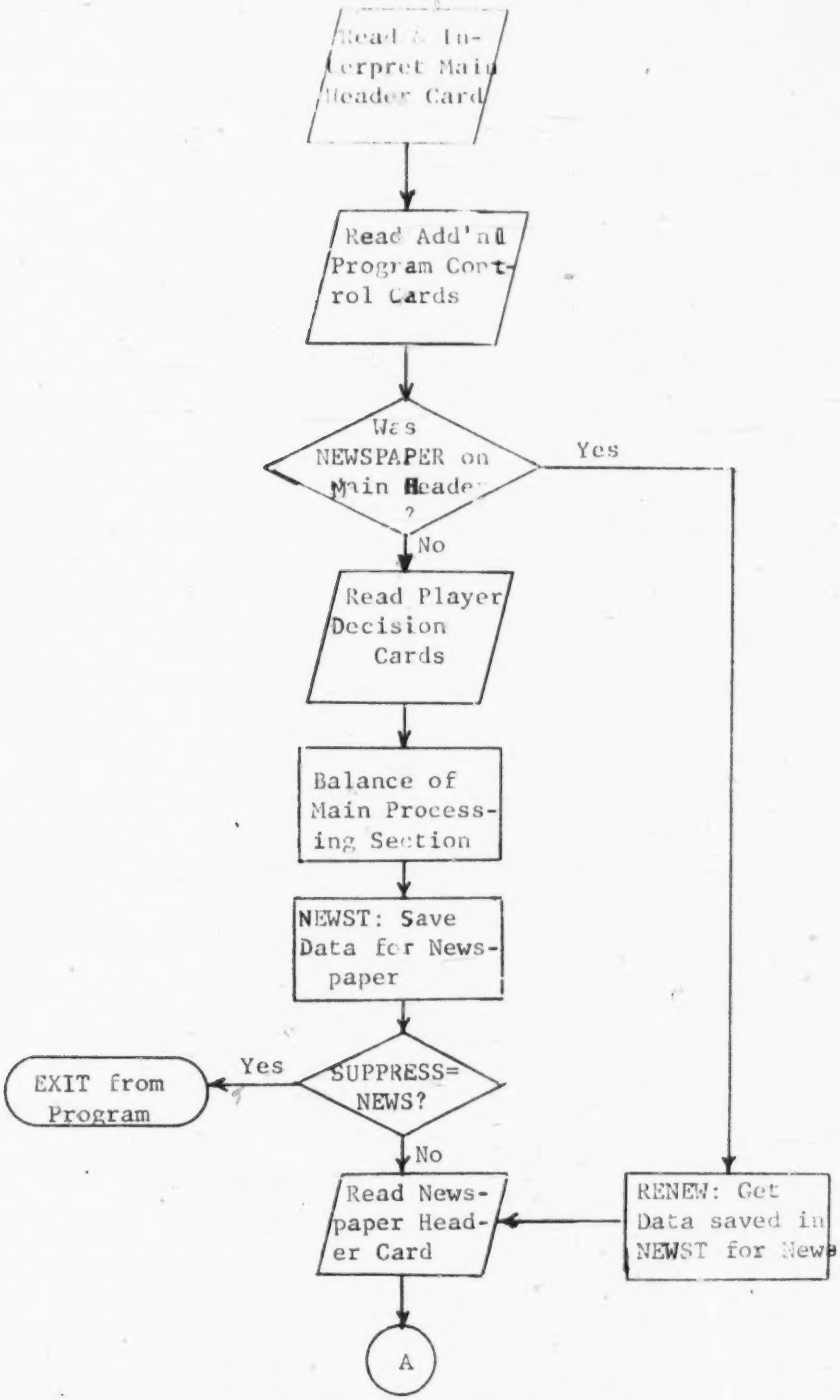
Examples of two-step data decks are included at the end of this chapter.

Job Control Cards -- input card set #1

The M.E.T.R.O.-APEX computer program and data bank are stored outside the main computer on a type of auxiliary storage known as a magnetic disk. Your

Figure 2

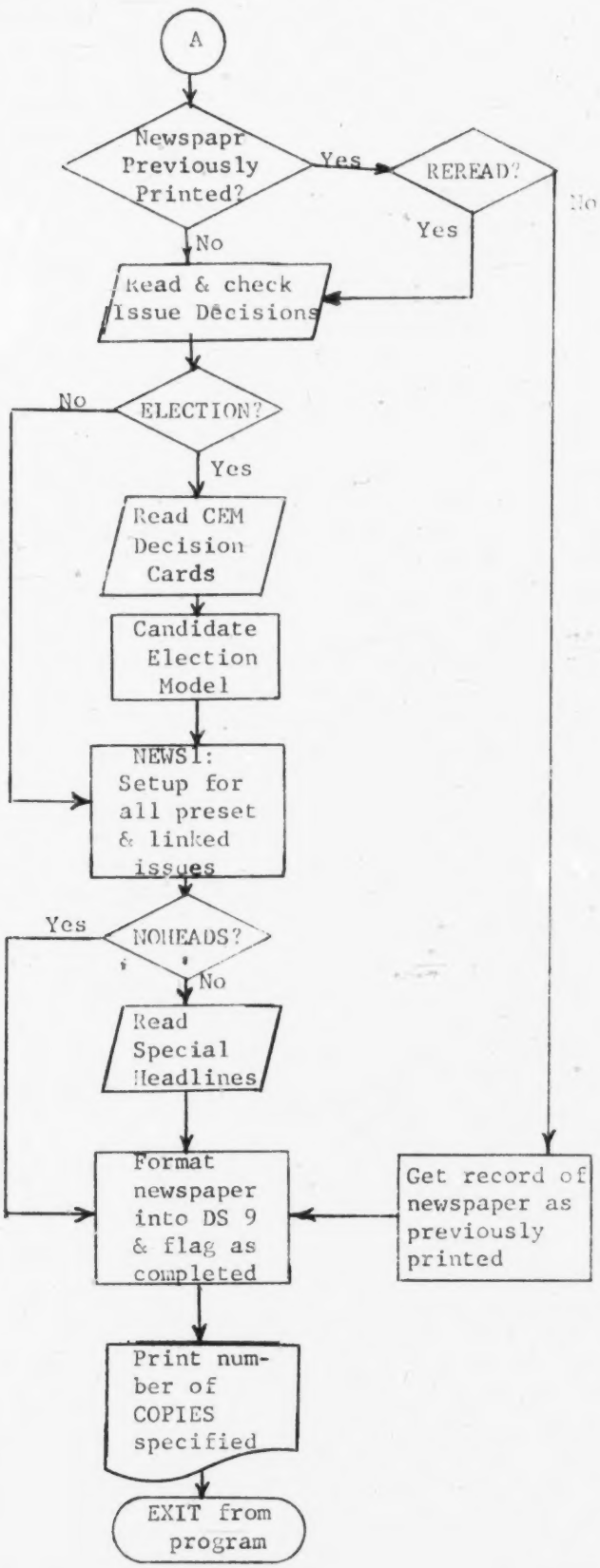
BASIC PROGRAM FLOW SHOWING NEWSPAPER REPEAT



(Continue next page)

Figure 2 (Cont.)

NEWSPAPER & ISSUE PROCESSING SEQUENCE



computer installation will probably have many of these disks available to the computer. They are capable of storing not only the M.E.T.R.O.-APEX programs and data but also many other programs and other data files. In order for any program to be executed by the computer it is necessary to transfer the program from its auxiliary storage into the main storage of the computer, tell the program where to find the data files it requires and finally instruct it to start operating. Depending on the model and size of IBM 360 you are running on, the M.E.T.R.O.-APEX program may be sharing main memory of the computer with one or more other programs, all running simultaneously. All of this activity is controlled by another program called the Operating System which resides permanently in the main memory of the computer and acts as "Big Brother" or traffic controller of the computer -- fetching programs from auxiliary storage, getting data for them and directing their output back to disk storage or to a printer.

This first set of cards in your input card deck is not part of the M.E.T.R.O.-APEX program but is the information required by the Operating System to control the loading and execution of the M.E.T.R.O.-APEX program (as well as information on how to charge you for the work done). It is written in a language all its own called "Job Control Language".

There are many elements of this Job Control Language which may be tailored to the requirements of a particular computer installation; therefore, as mentioned in the introduction to this manual, you should discuss this set of control cards with a systems analyst or other consultant at your particular installation before running M.E.T.R.O.-APEX the first time. The examples given here will work in most installations, but there are many options available, some of which will be discussed in more detail in Appendix B.

Figure 3

BASIC JOB CONTROL CARDS REQUIRED FOR MAIN M.E.T.R.O.-APEX PROGRAM

(Assumes only one team and records of all preceding cycles not saved)

```
1-10      11-20     21-30     31-40     41-50     51-60     61-70
1234567890123456789012345678901234567890123456789012345678901234567890
1 //jobname  JØB   accounting info.,name,etc.,REGIØN=120K,TIME=mm
2[//JØBLIB   DD    DSN=APEXLIB,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx]
3 //          EXEC  PGM=METAPEX
4 //FT02F001 DD    DDNAME=SYSIN
5 //FT03F001 DD    SYSØUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)
6 //FT06F001 DD    SYSØUT=A
7 //FT09F001 DD    UNIT=SYSDA,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210),
7b//                DISP=(NEW,DELETE),SPACE=(CYL,(2,1))
8 //FT14F001 DD    DSN=FILE14,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx
9 //FT17F001 DD    DSN=FILE17,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx
10//FT18F001 DD    DSN=FILE18,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx
11//FT19F001 DD    DSN=FILE19,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx
12//FT15F001 DD    DSN=FILE15,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx
13//FT16F001 DD    DSN=FILE16,DISP=ØLD,UNIT=2314,VØL=SER=xxxxxxxx
14//SYSIN    DD    *
```

Figure 3 shows the basic Job Control Cards for inclusion in set #1 of the input deck. Several items on these cards have not been explicitly specified and you will need to substitute values specific to your installation. (If you expect to be running M.E.T.R.O.-APEX frequently at the same computer installation, discuss with your installation representative the possibility of creating what is known as a "Catalogued Procedure" which would greatly reduce the number of Job Control Cards required. Details on suggested catalogued procedures are included in Appendix B.) The following is a card-by-card discussion of each card in the Job Control set:

Line 1: The "JOB" card will most frequently vary from one installation to another. In some cases the installation will supply you with pre-printed, pre-punched JOB cards which will include accounting information and/or other installation dependent information. Usually you will be able to add additional options to the card supplied.

If you are required to punch your own JOB card you must include the following:

a) Substitute for "jobname" an actual name up to 8 characters long beginning with an alphabetic character, such as MAINPROG, or MYJOB007.

b) If accounting information is not required, you must punch a comma preceding your name to indicate the intentional omission of accounting information. It is recommended that you include in the name field, the name of the person submitting the job to the computer.

c) Do not include any blanks or other special characters in the name field unless you begin and end the field with apostrophes; , ie., LASTNAME, or, 'C. PRATT',

d) The present 360/OS version of M.E.T.R.O.-APEX requires a region size of 120K bytes of main storage. This must be indicated on the JOB card as shown.

e) The maximum time that the program is allowed to use the central processing unit of the computer is usually set by the computer installation at between 30 seconds and 1 minute unless you specify otherwise on the JOB card. METRO-APEX will require more than this default time to process a complete cycle. Complete time estimates for all IBM 360 models is not available, however the time is approximately 8 minutes on a Model 50 and slightly less than 4 minutes on a Model 67. It is usually wise to specify a time limit slightly greater than you think will be required to make sure that your job is not abnormally terminated with only a little more time needed. We usually specify, TIME=10 for a Model 50 and TIME=5 for a Model 67. NOTE: Since the priority a computing installation will assign to a job will decrease as time increases above the default time, if you are only running the player input decision cards through to check for errors, which will require relatively little time, you should omit the TIME parameter for data checking runs only. Don't forget to include it on your JOB card after you have corrected errors and expect the program to run to completion.

There may be additional parameters, such as job class, etc., which will be required by your installation.

Line 2: (Optional) If your installation has allowed you to "catalogue" the M.E.T.R.O.-APEX main program, under the name METAPEX, in the System Library this card will not be required. The System Library is, however, usually reserved only for very frequently used programs and, in most cases, the program will not have been installed in this way. Therefore, the //JOBLIB card will be required to tell the system where to look to find the METAPEX program. You will need to replace the XXXXXX following VOL=SER= with the identification serial number of the disk pack on which the partitioned data set APEXLIB is located. Note here for further reference: VOL=SER= _____

Line 3: Punch exactly as shown.

Line 4: The data set reference number for card input data throughout the METAPEX Program is 2. (This is different than normal IBM 360 FORTRAN). This card informs the system of this fact.

Line 5: The printed output data set number used throughout METAPEX is 3. (This is different than normal IBM 360 FORTRAN.) The maximum line length throughout METAPEX is 121 characters (120 printed plus 1 carriage control). Check this DD card with your installation representative for possible local modifications. Point out to them that a complete cycle with all roles playing will produce from 120 to 150 pages of computer output; this may require an output class other than A.

Line 6: Required for system error messages.

Line 7: This data set is used only for printing the newspaper. To save time in printing multiple copies of the newspaper, the first copy is written into this temporary sequential file which is then rewound and copied n times to SYSOUT. Your installation may wish to modify the specification.

Lines 8 - 11: Data sets 14, 17, 18 and 19 contain the initial data base for METAPEX, constants used from cycle to cycle, and issue processing records. They are created by the FILEPRNT program (part of the M.E.T.R.O.-APEX system in APEXLIB) and are normally independent of the number of games being played. Normally all of these data sets will be stored on the same disk pack. Replace XXXXXX on each of these lines with the correct volume serial number on which they are stored at your installation.

Lines 12 - 13: Data sets 15 and 16 are the basic files containing variable data reflecting the changes in role accounts from cycle to cycle as game play progresses. At the beginning of the main section of the program both of these data sets contain exactly the same information. During the run of the main section of the program data set 15 is continually updated, so that at the end of the main section data set 15 contains the "current" state of the game (the starting point for going on to the next cycle) while data set 16 contains the state of the game before the computer run; i.e. at the beginning of the cycle just played.

From the above, it follows that if the next run of the program specifies going on to the next cycle, data set 15 is copied to data set 16 before the main program section. However, if you choose to repeat a cycle, data set 16 is copied to data 15 before the first main section. (More details of this cycle updating and/or repetition method is given in a later section of this manual.

For most game operations the two control cards shown will be sufficient with the insertion of the correct VOL=SER= number for your installation. For those game operatores who might be running more than one game run at the same time or who wish to save a record of all cycles played, the DSName parameter will need to be changed to be unique for each

team and/or each cycle. More details on this will be given later.

Main Program Header Card -- input card #2

This card is the first card read by the M.E.T.R.O.-APEX program and is always required. It identifies to the program the team number (required) and tells the program whether you are (1) starting a new game, (2) going on to a new cycle of play, (3) repeating the entire preceding cycle or (4) running only the issue and newspaper section of the program. You may also, on this card, specify a special team identification and/or special date to be printed on each page of the player output.

1. One, and only one*, of the following four keywords must be punched on the main header card:

   CYCLE=n,
   REPEAT=n,
   NEWGAME
   NEWSPAPER

   *There is one exception to the rule that only one of these keywords may be used. It will be explained later.

   Explanation: CYCLE=is used to indicate that you are going on to a new cycle. The new cycle number must be punched following CYCLE and it must be one greater than the preceding cycle.

   REPEAT=n indicates that you wish to repeat or restart a cycle. The value n which must be punched must be one greater than the cycle number stored in data set 16 (normally this is the same as the cycle you have just completed).

   NEWGAME indicates that you are restarting from the cycle 0 history; i.e. you are about to run cycle 1 for a new game run. (Note: NEWGAME is included for convenience. Actually NEWGAME, CYCLE=1 or REPEAT=1 all produce exactly the same result.)

NEWSPAPER is used if you wish to bypass the first main section of the program and skip directly to the issue and newspaper section.

2. The team number must be punched on the main header card as follows:

   TEAM=n,

   Usually the team number will be 1 (except for special cases discussed in the section on Running More Than One Game). The program checks the team number stored on disk against this number to prevent errors.

3. You may specify the date to be printed on each page of player output:

   DATE=mm/dd/yy

   The date specification is optional. If not specified the date used for the preceding cycle (or the last time DATE was specified) will be used. Note: you should always specify a date at the beginning of a game (when NEWGAME is used). The program will check the date given and, if correct, will convert it to alphabetic month and day of week for printing in the newspaper masthead.

4. You may specify a game run identification of up to 40 characters which will be printed on each page of the output in addition to the date. Usually this is specified only when starting a new game run; i.e. when running cycle 1. Unless specified the identification ' ***M.E.T.R.O.-APEX INITIAL CYCLE DATA ***' will be printed. Once a new team identification is specified, it will remain in effect for all following cycles until changed. The method of input is as follows: ' new team ident. ' Any combination of letters, numbers, special characters and spaces may be used, preceded and followed by apostrophes (') except apostrophes. You need not count exactly 40 characters. If you punch

fewer than 40 between apostrophes the program will fill the remaining spaces with blanks. However, if you punch more than 40 any additional characters will be ignored.

Rules for punching the main header card: All 80 columns of one card may be used. Punching may start in any column but may not be continued to a following card (you should never need more than 80 columns). Keywords must be separated by blanks, commas or, where indicated, by = signs. The end of a numeric field must be indicated by a comma except for the DATE field where a slash (/) may be used to separate the month, day and year fields. The four possible fields on the main header card may appear in any order. All of the following examples are correct:

TEAM 1, CYCLE 3, 'EXAMPLE OF TEAM IDENT', DATE 2/1/72

CYCLE 1, DATE 8/31/71, TEAM 1, 'THIS IS A NEW GAME $¢*L@# TIME 8:45'

REPEAT 2, TEAM 1

NEWSPAPER CYCLE 0, TEAM 1, DATE=10/27/71, 'PRINT CYCLE 0 NEWSPAPER'*

Note: The last card above is the one possible exception to the rule that only one of the first four keywords may be used on a card. If used, the words NEWSPAPER and CYCLE must appear in that order and the number must be zero; CYCLE 0 NEWSPAPER is incorrect.

Note in the above examples that either a blank or = may separate keywords and numbers. Blanks must never be used within keywords, although only the first four characters of any keyword need be punched.

*The cycle 0 newspaper is the newspaper that precedes the first preset cycle 1 decision sequence. It is used primarily to initiate linked issues in subsequent newspapers and may be printed for distribution to the players to give them the background information on issues appearing in the cycle 1 newspaper.

Caution: This option should be exercised prior to the start of a new run only. Printing the cycle 0 newspaper destroys the record of the last cycle run for the designated team and resets the file for starting a new run. Normally, this option should not be used.

Additional Program Control Cards: -- input card set #3 (optional)

Additional program control cards are used to inform the M.E.T.R.O.-APEX program of non-playing roles, suppress printing of output, cause printing of some output not normally printed (usually for diagnostic purposes), and inform the program of some special newspaper options.

The additional program control cards consist of seven possible keywords followed by an equal sign followed by a string of options separated by commas. Each card must start with a keyword= and must not contain any embedded blank columns. The same keyword may be used on as many cards as necessary but the list of options may not continue from one card to another. Only one keyword may be used per card. Keyword cards may appear in any order. If contradictory options are included, the last one read remains in effect.

keyword	options	
NOPLAY=	ALLDEVELOPERS	
	ALLINDUSTRIALISTS	
	DEVELOPER (i)	(i can be 1 through 7)
	INDUSTRIALIST (i)	(i can be 1 through 7)
		Note: INDUSTRIALIST (4) and (7) are legal but unnecessary.
SUPPRESS=	DEVELOPER (i)	(i can be 1 through 7)
	INDUSTRIALIST (i)	(i can be 1 through 7)
	POLITICIAN (i)	(i can be 1 through 5)
	PLANNER (i)	(i can be 1 through 5)
	APCO	
		Note: IND(4) and (7) and POL and PLAN (2), (3) and (4) are legal but unnecessary. The default is automatically SUPPRESS for

keyword	options	
SUPPRESS=	DISTRIBUTION	
	SUMMARY	
	DATA	
	ALL	
	NEWS	
PRINT=	POLITICIAN(i)	(i can be 1 through 5)
	PLANNER(i)	(i can be 1 through 5)
	TOMMSTART	
	TOMMEND	Note: POL and PLAN (1) and (5) are legal but unnecessary. They are printed by default unless suppressed.
DIFFUS=	ALL	
	SPRING	
	SUMMER	
	FALL }	Synonymous
	AUTUMN }	
	WINTER	
ELECTION	none	
BYPASS=	CYCLEONE	
	LINKS	
	NATIONAL	
	LOCAL	
GEM=	DIAGNOSTICS	
	RESULTS	

(Most of the above keywords and options may be abbreviated for ease in keypunching. The minimumnumber of characters which must be used in each case is underlined.)

The effect of each of the above is as follows:

NOPLAY: All seven developer roles and industrialists 1, 2, 3, 5 and 6 are considered playing by default unless the program is explicitly informed otherwise by use of a NOPLAY card(s). It is advisable to specify

non-playing roles since their omission will save a small amount of computer time. If not played in any cycle, the status of the files for a particular role remains exactly as it was the last cycle played, including taxes and loan payments owed and property holdings. If a role is not indicated as nonplaying and no loan or tax payment decisions are input for that role, the program will assume it was an intentional nonpayment and will assess delinquency payments and other penalties against that player's files which could create confusion if the player had merely missed a cycle and wished to resume play two cycles later.

Notice the ALLDEV and ALLIND are available as a short-cut in the event you wish to suppress all of either or both types.

SUPPRESS: Output for all of the developers and industrialists will automatically be printed if that role is playing unless you specify otherwise with a SUPPRESS card(s). (Notice that you need not specify both NOPLAY=D(1) and SUPPRESS=D(1); the NOPLAY automatically assumes no output.)

Central City and County Politicians and Planners (POL(1), POL (5), PL (1), and PL (5) respectively) and the Air Pollution Control Officer's output will normally be printed unless suppressed. DISTRIBUTION refers to three tables in the output -- a bar graph of land distribution and tables of developed and vacant property and prices. SUMMARY refers to several property and value tables, population and capital plant index summaries. Note that some tables usually given to the planners are categorized and printed among the SUMMARY tables; suppressing SUMMARY would suppress all of these.

Normally all of the main player decision cards (input set #5) are printed as they are read. The SUPPRESS=DATA option will suppress this complete listing. All cards with any errors will still be printed along with a diagnostic message indicating the type of error. Since it is usually a good idea to check the input card listing for unusual decimal values, even though no program detectable error is indicated, this option should normally not be used except to suppress the printing of preset cycle 1 data cards or input lists of cycles being repeated.

The SUPPRESS=ALL option will result in the suppression of all player output for the roles which preceded ALL in the list of options. It would normally be used only if you are repeating cycles with very minor changes for the purpose of file modification and no output was desired. Note that the ALL option does not suppress the newspaper section of the program.

SUPPRESS=NEWS is the mechanism for running only the first main section of the program as suggested at the beginning of this chapter. If this option is used only input card sets 1 through 6 need be included in the computer input deck. Even if this option is used, the data required by the issue and newspaper processing section from the first part of the program is saved in the newspaper files and need not be recalculated.

PRINT: Normally the output for simulated jurisdictions 2, 3 and 4 is not printed as it serves little use in the present version of the game. The printing indicated for planners 2, 3, 4 is only the recommended capital projects for simulated jurisdictions 2, 3 and 4. Since normally

no projects are recommended for these jurisdictions (county financed projects in those areas are shown as planners recommendations to the county) the output is not normally printed.

TOMMSTART causes the printing of the initial distribution of population, employment and land use prior to the reallocation performed by the TOMM model. It is usually nothing more than a reiteration of summary data from the last cycle and not in a form which is easily read by the players, hence is usually printed only for diagnostic purposes, which is rare. TOMMEND, like the start picture is for diagnostic purposes only and is not normally printed.

DIFFUS: Any of these options print a detailed table of the levels of all pollutants in all of the analysis areas. The option was originally incorporated in the program for diagnostic purposes and the output is not easily read by the average player. ALL prints this information for all four seasons, but you may obtain it for any one or more of the seasons by so specifying.

All of the preceding options apply to the first main section of the program. The following options apply only to issue, candidate election model and newspaper processing but, if used, they must be included in input card set #3. They may be included even if the newspaper is being suppressed; they will simply be ignored.

ELECTION: This keyword card, which has no additional options, must be included to inform the program to process a candidate election. Unless this card is included, input card set #9 (Candidate Election Model decision cards) will not be read, even if included in the input deck.

CEM= DIAGNOSTICS will produce results unintelligible to anyone but the designer of the model. RESULTS prints a special analysis of the election result for each candidate; that is, for each election, the percent attained by the incumbent will be broken into parts reflecting EOP performance, past performance in office, party support and the random factor (see Vol. 1, Chapter 4). It is usually used only for diagnostic purposes. These results are printed preceding the newspaper. The omission of CEM=RESULTS does not prevent the election results from being printed in the newspaper; the final results will be printed in the newspaper if an election is held.

BYPASS: Normally this option will not be used.

Preset decisions for the three issues in the "cycle 0" newspaper are stored in disk "file" 237 and are normally used when printing the newspaper at the end of cycle 1. Notice that when running cycle 1 the Issue Decision Card (input set #8) must not be included in the deck unless BYPASS=CYCLEONE is specified. If the game director wishes to alter the contents of the first newspaper given to the players, he can specify BYPASS=CYCLEONE in which case an Issue Decision Card with the decisions for the cycle 0 issues 1, 42 and 50 must be included in the cycle 1 data deck.

The three issues in the "cycle 0" newspaper are:

#42-- (county) Runway expansion for airport, project 109. This is approved in the cycle 1 issue decisions. The main preset decisions for cycle 1 include a CP card for project 109 and BS card for bond number 111 to finance it. If this decision is changed the CP card and BS card should be removed from the cycle 1 data deck and new County Politician output (different than that in the player manuals) should be produced.

#1 -- (Central City) Build new City Hall, project 86. This is also approved in the cycle 1 preset decisions and a CP card for project 86 and BS card for bond #102 are included in the preset main cycle 1 decisions. As above, if this decision is altered the CP and BS card should be removed and new City Politician output run.

#50 -- Annexation of Maple Grove Area, part of Analysis Area 12. This area is assumed annexed to the central city in cycle 1 and creates a demand in the cycle 1 newspaper for needed projects to service the area.

BYPASS=LINKS will allow issue decisions entered for issues to be printed in the newspaper but any issues which would be generated by linking from a decision on one of the preceding cycle issues will not be printed.

BYPASS=NATIONAL will suppress all preset state and national headlines for that cycle. (A list of the preset state and national headlines may be obtained by running the FILEPRNT utility program and printing file 204.) These headlines are preset to give an indication of the national and state economy and social problems. You could simulate your own national and/or state headlines by inputting them as special headlines with a code of 1.

BYPASS=LOCAL will suppress all preset county and local issues. It will not suppress issues linked to from preceding cycles, capital plant headlines, low per-capita budget generated issues or business page issues.

As many Additional Program Option Cards as required may be included in the input deck. The following are some examples of correct Program Option decks:

```
ELECTION
SUPPRESS=DEV(4),APCO
NOP=I(2),I(5),D(5)
PR=POL(4)
RUN
```

The following set is exactly equivalent to the above:

```
SUP=D(4)
SUP=APCO
NOPLAY=DEVELOPER(5)
NOPLAY=IND(2)
NOPLAY=INDUSTRIALIST(5)
ELECTION
PRINT=POLITICIAN(4)
DATA
```

In the following example of contradictory options, Politician 5 would be printed since the last card read is the one in effect:

```
SUPPRESS=POL(5)
PRINT=POL(5)
RUN
```

The RUN (or DATA)Card - input card set #4:

The card is actually an additional keyword, but is shown as a separate input card set because it must be included in the input deck whether or not you have used any additional option cards. It indicates to the program the end of the Additional Option Card set and allows the program to continue on to the Player Decision Cards. You may use either RUN or DATA, not both.

From Worksheets to Cards (Main Player Decision Cards, Input Set #5)

Prior to the actual punching of the main input cards, the operating team must transfer and translate the decisions on the player's worksheets to coding forms. The player decisions are sometimes expressed verbally rather than in number code and appear, along with assorted extraneous

entries, on the worksheet. To facilitate keypunch operations, it is advisable for the operating team to translate the verbal decisions to numeric code and enter that code in the proper location on a key punch form.

In certain instances, players may find the format of the worksheets time consuming, unproductive and restrictive. In those cases, the keypunch forms themselves can be used by the role advisors to direct the players through their decisions. In the extreme case, the players can go directly to filling in the keypunch form bypassing both worksheet and role advisor.

Copies of the keypunch forms for each role are included in this chapter for reference. (Additional copies are available with the packet of worksheets and forms that can be purchased along with this manual.) The starred items on the worksheet have to be entered on the keypunch form. The coding for those items entered in verbal rather than numerical form is explained on the page following the respective coding forms.

An example of the translation process will best demonstrate the procedure for filling in the coding form. (After a team of role advisors gains experience in filling in key punch forms, the expected time duration for the transfer and translation process is about five minutes per role.)

Once the decisions are entered correctly on the key punch coding forms, the input cards must be punched. It is assumed that the reader is capable of using the key punch -- if not, a short introduction should be all that the reader needs. (The key punch is little more than a high powered typewriter, one designed for putting holes in IBM cards rather than for printing letters on paper.)

Data is entered on cards to conform with an arrangement in accordance with program specifications. Two distinct arrangements, called formats, are used. The first and most common allows up to thirteen entries on a card. The first takes up two columns, the next six take up four apiece for a total of twenty-four, and the last six take up nine each, filling the remaining fifty-four. Each grouping of columns is called a field. The first two columns are reserved for an alphabetic code of two letters. Each card must have an alphabetic code. The next six fields are reserved for integer entries. These must be right justified; that is, the entries must appear as far to the right in each field as possible. For example, the number 12 would be entered in columns 5 and 6 on a card if it belonged in integer field 1. The remaining six fields are reserved for real number entries (numbers with decimal points). The entries may appear anywhere in the field but a decimal point must be punched in the appropriate location.

A Word of Caution: Occasionally a field will require punching a minus (-) sign. When punched, the sign must be the first non-blank character in the field. (Many keypunch operators prefer to fill in the left-hand side of integer fields with zeros instead of blanks; e.g. BS00010025 instead of BS 1 25. This has certain speed advantages when using a drum card, a discussion of which follows.) If you are punching leading zeros, the proper form for a negative number is, for example, -001 not 00-1. The latter form will result in the value zero being stored, not -1.

Each card corresponds to one line on the key punch coding form. The two letter code adjacent to each table is the two letter code in the first

field. The designation in parentheses of "I-*" indicates that the entry belongs in the *th real number field. (For FORTRAN buffs the input format is (A2, 614, 6F9.0) with integers right justified.)

The second format is used only twice, for PI and EM cards. The two letter code is handled exactly as it is above. However, the remainder of the card is partitioned into 15 four column integer fields; that is, all entries are integers and all fields are four columns wide. Each entry goes in the field indicated by the I-* designation on the key punch form and each is right justified. (For FORTRAN buffs the input format is (A2, 15I4) with integers right justified.)

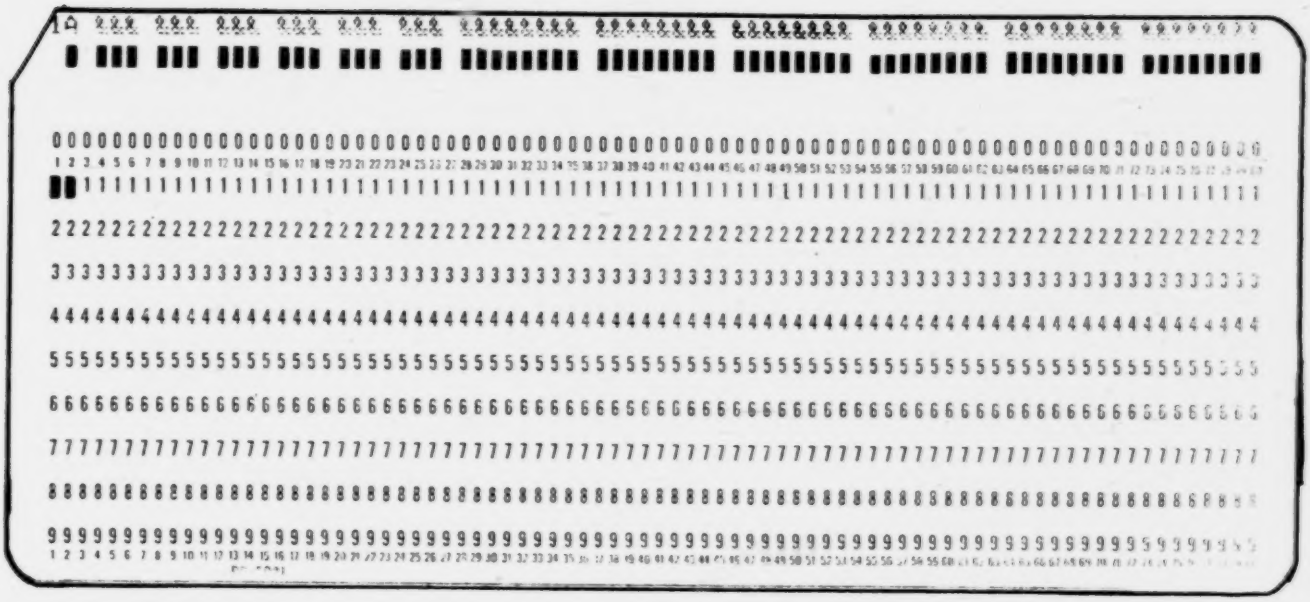
FOR THE IBM 360 VERSION OF M.E.T.R.O.-APEX THERE MUST BE TWO PI AND TWO EM CARDS ADJACENT TO EACH OTHER IN THE INPUT DECK. They may be exact copies of each other or the first of the pair need have only the proper alphabetic code in the first two columns. (The first occurance of the code tells the program to read the data from the following card in the different format.)

The key punch procedure is greatly simplified by the proper use of a drum card. A drum card is a high powered tab stop for a keypunch -- it not only jumps to the right column; it sets the numeric or alphabetic (upper or lower) case as well. Thus, by using a drum card, letters are placed automatically in the first field and numbers are punched in all other fields. Note, use the numeric decimal point in all cases. Also, use caution when using the skip key (the analogue to the tab). When the machine is resting in the first column of a field, depressing the skip key advances the field. (The key punch shifts to the next field.)

A drum card for M.E.T.R.O.-APEX is made by placing a 1 in column one of an IBM card followed by an A. The first column of each remaining field is blank with ampersands filling the balance. (See Figure 4.)

Figure 4

KEY PUNCH DRUM CARD FOR
MAIN PLAYER DECISION CARDS



For easy reference, the following is a chart of where each field begins on the card:

Field Field	Begins in column	Field	Begins in column
I-1	3	F-1	27
I-2	7	F-2	36
I-3	11	F-3	45
I-4	15	F-4	54
I-5	19	F-5	63
I-6	23	F-6	72

Note: The ampersand (&) is punched in what is frequently called the '12-row of the card.

The XX card -- Input card set #6

Once all of the main player decision cards are complete, an additional card with the letters XX punched in the first two columns must be added to the deck immediately following the last main player decision card. The XX card serves two purposes: (1) it signals the computer program that the last main player decision card has been read and (2) it informs the program what action to take after having read the player decisions. Normally the I-1 field (columns 3-6) is left blank in which case the program will continue on with the normal processing sequence if there were no program detectable errors in the player decision cards but it will stop if the program did find errors in the player decisions.

You may alter this normal action in one of two ways:

(1) You may elect to have the program stop unconditionally after reading the player decision cards, in which case you must punch a negative number in the I-1 field (usually -1 in cols. 5 and 6).

(2) In very rare cases you may elect to have the program continue on even though it did detect errors in the player decision cards (for example, if you were starting a new game without playing all of the developers and/ or industrialists and did not want to remove all of the preset decision cards for the non-playing roles from the cycle 1 data deck). Other errors will usually cause the card in error to be ignored and may result in unforeseen errors in the final output. Therefore, this option is to be used with great care. If you do elect to continue on despite errors, punch a positive number in the I-1 field of the XX card.

Newspaper Header Card -- input data set #7:

The Newspaper Header Card is always required unless SUPPRESS=NEWS is specified in the additional option cards (input set 3). It is required even for cycle 0 and cycle 1 newspapers since it informs the program of the number of copies of the newspaper to be printed and some other possible options.

The following are the possible keyword options on the Newspaper Header Card:

[COPY=n, / COPIES=n,]	[IGNORE]	[NOHEADS]	[REREAD]

At least the first four letters of each keyword must be punched. There must not be any blanks within a keyword, but each keyword must be separated by either a comma or a blank. The comma following the number of copies is required unless it is the only or last option on the card.

COPY= or COPIES=: Either form of this keyword may be used, followed by the number of copies of the newspaper you wish to have printed. If this keyword is omitted the default is one copy.

IGNORE: Once the entries on the Issue Decision Card are accepted as valid, the program verifies that the issue list is consistent with the linkage from the preceding cycle. (Each cycle, just prior to printing the first newspaper, the program prints a list of all issues that will link to other issues in the next cycle. This list is saved and the program checks that the issue decisions input for that next cycle include decisions for all those issues that required resolution to determine the nature of the linkage.) If some issue requiring resolution to direct the linkage to related issues is not included in the issue decision sequence, the following is printed:

**MISSING DECISION FOR ISSUE nn (Other error messages are listed in Chapter 3.)

Normally, if missing issue decisions are found, the program will stop, the errors must be corrected and the newspaper section of the program restarted by specifying NEWSPAPER on the Main Header Card.

The use of the option IGNORE on the Newspaper Header Card will cause the program to continue on despite the missing decisions. Any further linking for those issues will be ignored.

NOHEADS: Since players almost always submit special headlines to be printed in the newspaper, the program assumes that Special Headline Cards (input set #10) will be read unless NOHEADS is specified on the Newspaper Header Card. Notice that, although you would not normally do so, special headlines can be read for both the cycle 0 and cycle 1 newspapers. Therefore IGNORE is usually specified on the Newspaper Header Card when printing a cycle 0 newspaper and frequently when running the cycle 1 preset decisions.

REREAD: Once the program has printed one copy of the newspaper for any cycle it "remembers" what was printed. Any number of additional copies of the newspaper may then be printed without rerunning the main section of the program and without reading any of the newspaper and issue decision cards (input sets 8, 9 or 10) simply by specifying NEWSPAPER on the Main Header Card and the number of copies on the Newspaper Header Card.

Specifying the REREAD option on the Newspaper Header Card allows the operator to rerun the newspaper for a cycle with different alternatives for some of the issues, different candidate data (if ELECTION is specified), different special headlines, or any of the other options possible (e.g., bypassing state and national headlines). In other words, REREAD forces the program to read input sets 8-10.

The ability to print additional exact copies of the newspaper, as mentioned above, may be useful if the operator wishes to check the printing of one copy before deciding whether to print many additional copies. For example, for some fiendish reason known only to the game director, he may want to be sure that some candidate wins or loses in the election. If, the first time through, the election result is not as desired, the operator can further bias the election results by changing the party support higher or lower and REREADing the decision cards. If the result was as desired, sufficient exact copies may be printed to pass out to players.

Notice that if one copy of the newspaper was not printed, due to a fatal error in one or more of the input card sets 8, 9 or 10, the program will assume that these cards are to be reread.

Issue Decision Card(s) -- input card set #8:

An issue decision card contains one or more groups of numbers separated by commas. Each group of numbers gives the program information about one issue appearing in the newspaper that was printed at the end of the preceding cycle. The information is comprised of

two required entries; the issue number and the alternative acted upon by the deciding agent (a politician or the elite opinion poll). If the alternative action required that a project be funded, the bond or millage number associated with that project may be given.

The format for issue decisions is as follows: iiia(bbb), iiia, iiia (bbb), . . . iiia*

Where: iii is a 1 to 3 digit issue number
a is a single digit indicating the alternative chosen on that issue,
(bbb) is an optional 3 digit bond/mill number

The card is "free-format"; that is, punching may begin in any column, and blank columns are ignored and may be used for spacing to increase legibility. Each group of numbers for a single issue is separated by a comma. However, the last group must be terminated by an asterisk (*), not a comma. A maximum of 40 issue decisions may be input in any single cycle. If one card is not sufficient for all issue decisions, decisions may be punched on any number of additional cards. The terminating asterisk appears only after the last decision on the last card of the series. The last character on any one card must be either a comma or an asterisk. Thus, a number group for any one issue must appear on only one card -- it will not be continued from one card to the next.

The following is an example of two issue decision cards:

```
1692,  1 1,

   56 1  (103),    28 3 *
```

This would be interpreted as follows:

Alternative 2 for issue 169, alternative 1 for issue 1, alternative 1 for issue 56 subject to the passage of bond 103 (otherwise, if bond 103 failed, the alternative would be changed to 2 by the program), and alternative 3

for issue 28. Note that all of the above could have been punched on one card; the two cards were used only for illustration. A key punch for issue decisions (see next page) is included in the packet of forms for use by the operator.

In each cycle, the alternative selected for every issue appearing in the preceding cycle's newspaper must be entered as part of the issue decision cardset. Use of the elite opinion poll wall chart tally sheet described in Volume I, Chapter 3 of this kit is extremely useful in keeping tabs on issues and issue outcomes.

Candidate Election Model decision cards -- input card set #9:

This input card set must be included, and may only be included, if you specified ELECTION in the Additional Program Control Card set.

There are up to 16 candidate election data cards in an election year. A card must be punched for each incumbent politician running for office. (Up to eight incumbents -- three central city politicians, one county commissioner representing each outlying jurisdiction and two county commissioners representing the city -- can run each cycle.) For each incumbent running for office, a card must be punched for either his simulated or game opponent.

The following format is used for punching candidate election data cards.

Cols. 1-2 Candidate Number. A candidate number must be entered on each card. Incumbents should be assigned numbers 1-8 according to the following schedule:

1= Central city politician, ward 1.
2= " " " " 2.
3= " " " " 3.
4= Suburban county commissioner
5= Township 1 (jurisdiction 3) county commissioner
6= " 2 " 4 " "
7= Central city "at-large" representative on county board.
8= " " " " " " "

Corresponding numbers from 9-16 should be assigned to their opponents so that each matched pair are given numbers 8 apart; that is, opponent 12 runs against incumbent 4, etc. Do not punch cards for opponents if there is not a card for the corresponding incumbent.

Col. 3

a) For incumbents (01-08 in columns 1-2) this column indicates whether the opponent is real or simulated:
   1 = gamed opponent (real)
   2 = simulated opponent
b) For gamed challengers (09-16 in columns 1-2) this column should be blank. For simulated challengers this column designates the opponent type:
   1 = Conservative
   2 = Moderate Republican
   3 = Labor Liberal
   4 = New Left Liberal
   5 = Working Class Conservative
   6 = Ultra-Conservative
   7 = Business Candidate
   8 = Black Candidate
   9 = Reform Candidate

Cols. 4-8

Party support. This entry is included only for incumbents. It is the major discretionary variable under the control of the operator. Some number between 1 and 60 is entered designating the percentage of the vote, up to 60%, going automatically to the incumbent. The entry can be determined by apportioning the 60% in accordance to a hand vote of gamed players, operator's whim or any other suitable method. If the entry is left blank, the machine sets the appropriate variable to 30. A decimal point must be punched in this field.

Cols. 9-28

Candidate Name. A name up to twenty characters long is entered here to be printed with the election results in the newspaper. Names should be entered for simulated as well as gamed opponents. For example, a conservative might be called Richmond L. Stuffshirt.

29-80

Issue Number, Candidate Stand on Issue, and Candidate Stress on Issue. The form follows that for issue decision cards -- iiia(bb) -- where iii is a 1 to 3 digit issue number, a is the alternative chosen and bb is the percent of campaign funds spent stressing the stand taken on that issue. These entries are free format and up to 40 issue decisions may be entered. Additional cards may be used as needed with the continuation card starting in column 1.

An asterisk is used to designate the end of the sequence for a given candidate. Entries should be made for only those issues for which the candidate casts a vote. Note that these columns should be left blank for simulated opponents. Also, for unstressed issues (issues which would have zero in parentheses) the parentheses may be omitted. Stress percentages must be entered as whole numbers; decimal points may not be included within the parentheses.

One blank card must follow the last candidate card to terminate this section of the input. The following is an example of a card for an incumbent candidate:

```
01125.00ROY MILLER      1691(20),0012(5),0563,0423*
```

Incumbent 1 has a real opponent named Roy Miller. Twenty-five percent of the party allocation has been assigned to him. He has voted for alternative 1 on issue 169 and spent 20 percent of his campaign funds stressing that stand, etc.

Special Headline Cards -- input card set #10:

Special headline cards are assumed part of the newspaper input sequence unless NOHEADS is specified on the Newspaper Header Card.

This input section contains special headlines written by the operator, a newspaperman or players in the game for publication in the News. These cards, like the others, must be punched to conform to program specifications. Each line of a headline appears on two cards. The first column of the first card of each pair must have a code describing the type of headline. The possible codes are:

1 = Special Headline (lead article)
2 = National Headline
3 = State Headline
4 = County Headline
5 = Central City Headline
6 = Suburb Headline

7 = Township 1 Headline
8 = Township 2 Headline
9 = Business Headline

The rest of the first card, columns 2-80 contain the first 79 characters of the line. The first 41 columns of the second card contain the last 41 characters of the same newspaper line. There can be a maximum of 118 lines in a given cycle.

The cards must be ordered so that the codes in column 1 are in ascending order from 1-9. To skip a line between headlines in a single section (with the same code in column 1), one of the 118 lines must be left blank; that is, the appropriate code should be entered in column 1 of a card but the remainder of the card and a second card should be left blank. The program automatically handles the placement of each section; therefore, blank cards are not needed between sections (between headline cards with different codes in column 1).

Delimiter Card -- input card set #11:

This card, which contains nothing but /* in card cols. 1 and 2 should always be included as the last card in the input deck. It serves two purposes; (1) it signifies the end of special headline cards, if present, and (2) signifies the end of the total input deck.

SUMMARY

In summary, the data deck required to run a cycle from start to finish is pictured in Figure 1. It is not necessary, as noted at the beginning of this chapter, to run the entire program at one time. Therefore all of the input card sets shown in Figure 1 need not be included

in every deck submitted to the computer. The following examples show subsets of the complete data deck which may be used for certain purposes.

Figure 5

INPUT DECK FOR ONLY MAIN SECTION OF PROGRAM

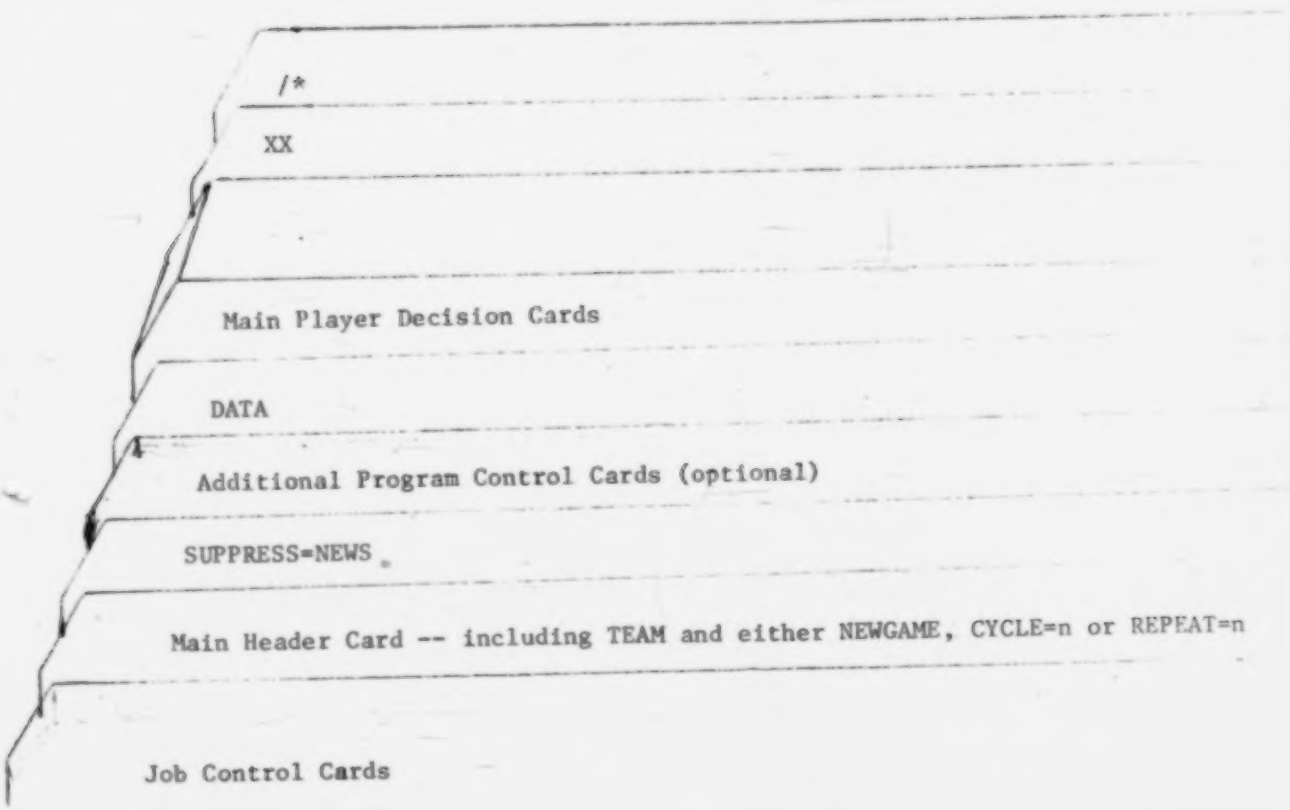


Figure 6

INPUT DECK FOR PRINTING NEWSPAPER

(If Newspaper Suppressed Previously)

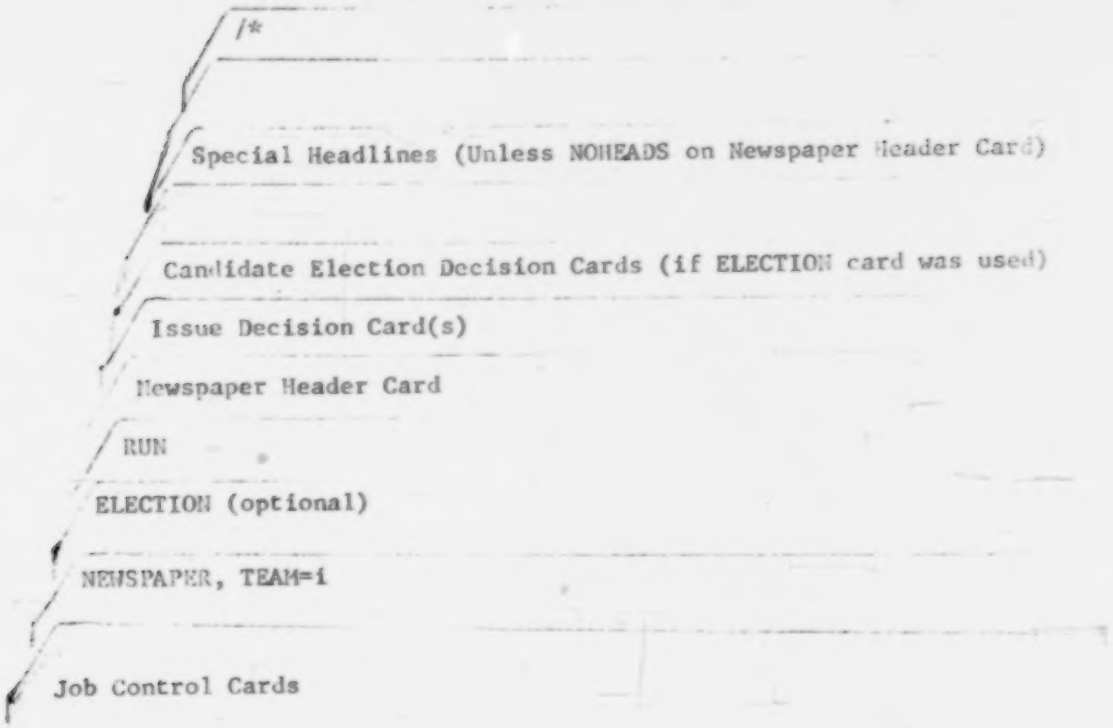
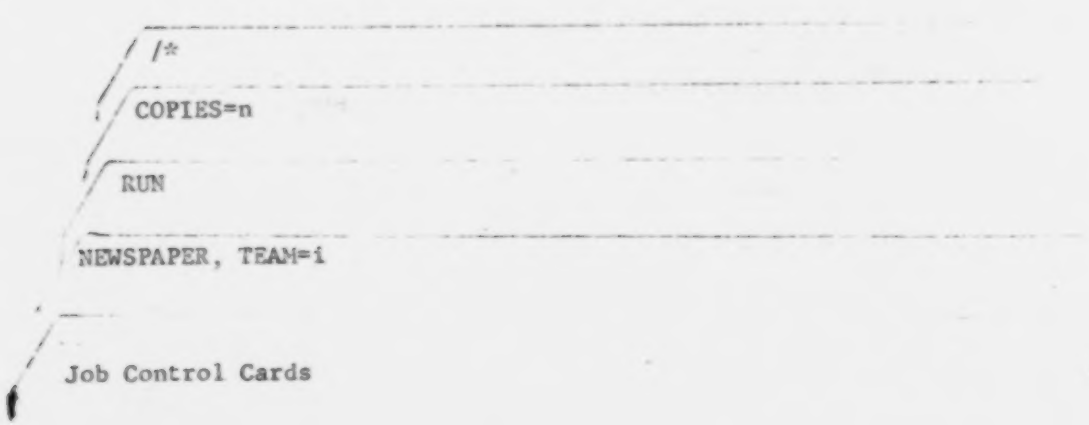


Figure 7

INPUT DECK FOR PRINTING ADDITIONAL EXACT COPIES OF NEWSPAPER
(Assume at least one copy has already been printed)



Chapter 3

ADDITIONAL GAME DIRECTOR INPUTS

In addition to the operational controls and player decision discussed in the preceding chapter, there are a number of additional options available to the game director or operator for changing the values used in some of the game models and otherwise manipulating the outcome of the game for a particular cycle. The mechanism for introducing these options into the game is by means of cards which are prepared in the same format as and are included in the input deck with the main player decision cards (input card set #5). Since they are not usually player decisions, space for recording the values is not usually included in the player worksheets and standard keypunch coding forms. Instead they are recorded as part of the "Instructions to Computer Operator" form. (Notice that two of these inputs are additional fields punched on the Player Decision Cards, although not on the player worksheets.)

Change U.S. Exogenous Employment Growth Rate:

This value is the percent change in exogenous employment, both industrial and office (bureaucratic), except for the gamed industries, state government employees and the university employment in a cycle. It is preset at the actual values for the years 1957 through 1966 unless changed by the game director. A change in this value will have an effect on the growth of the entire gamed area. The precise change is not completely predictable, since growth is also affected by new industries entering the area (see Forcing Exofirms below), additional exogenous employment in specific analysis areas (see Additional Exogenous Employment

below) plus a small random factor. Changes in this value will also, obviously, have an effect on the M.E.T.R.O.-APEX area unemployment rate -- the average unemployment changing in inverse ratio to the employment growth.

The Employment Growth Rate may be changed by including in the Player Decision deck a card with EG punched in columns 1 and 2 and the new percentage value in columns 27-35 (be sure to punch a decimal point).

Unless changed by use of the EG card the following are the preset U.S. Employment Growth Rates used by the program for each cycle:

Cycle	Percent
1	-0.9
2	0.5
3	2.5
4	1.7
5	0.2
6	1.6
7	1.4
8	2.3
9	2.6
10	0.8

Forcing Exogenous Industries (Exofirms) into Specific Areas:

Each cycle the newspaper lists a small group (from 3 to 6) of exogenous industries that wish to locate in the METRO-APEX area. Each

of these industries has a preferred location or locations (listed in order of preference) plus one or more conditions which must be satisfied by the politicians and/or developers in order for them to locate in the community. These conditions may be construction of streets or sewers, rezoning of land for industrial use or investment of money by the developers.

The players are sometimes confused as to the specific actions they must take. As far as street and/or sewer construction is concerned, the program does not look for specific projects -- it simply checks to see whether the total dollar amount of all projects of the correct category (streets or sewers) being allocated to the desired analysis area is equal to or greater than the amount required by the exofirm (notice that they must be AA-wide projects; the program does not check for money spent in the ward or jurisdiction). The total amount of all developers' and industrialists' investments in that exofirm must be equal to or greater than the amount specified in the newspaper. Note that more than one developer and/or industrialist may invest in the same exofirm and that the amount invested may be more than required. The rezoning requirement will be met if the total number of acres rezoned to vacant industrial (vacant type 4) is equal to or greater than the number of acres required. The old land use type is immaterial and the total may be made up of several small parcels or one large one. The program considers all such parcels as contiguous. The game developers should be advised that the exofirms which do enter will not purchase vacant industrial land from them but only from the general market.

They should also be advised that they cannot rezone land to vacant industrial and sell that land to the market in the same cycle.

Sometime the players will form a scheme to create an industrial park in some location and request that the exofirms locate in that area. Other times the game director will wish to force in an exofirm despite the fact that all player requirements have not been met, or he may wish to locate the exofirm in some area other than those listed in the newspaper. The "Exofirm Force" card is the mechanism for accomplishing either of these goals. The format is:

Cols. 1-2	XF
Cols. 3-6	Exofirm number, right justified.
Cols. 7-10	Analysis area into which the firm is to be forced. If zero or blank the firm's first choice will be used.

Notice that there must be one card for each exofirm to be forced.

Additional Exogenous Employment:

Another method the operator or game director can use for changing the distribution or magnitude of employment in a specific area is by means of the EE (Exogenous Employment) card. By this means he can create larger growth than available with most preset exofirms. The format is as follows:

Cols. 1-2	EE	
Cols. 3-6	Analysis area in which to locate	
Cols. 7-10	Type of employment:	1 = Industrial (manufacturing) 2 = Office (bureaucratic)
Cols. 27-35 (F-1)	Number of employees (include decimal point)	
Cols. 36-44 (F-2)	Number of acres required (include decimal point)	
Cols. 45-53 (F-3)	Dollars of added tax base (include decimal point)	

Entering Background Pollution Levels for Each Season:

The game director may wish to create an air pollution crisis by adding a fixed quantity of each contaminant throughout the region in one or more seasons. The mechanism is by means of the BP card, the format for which follows:

Cols. 1-2	BP	
Cols. 3-6	Season code number:	1 = winter 2 = spring 3 = summer 4 = fall
Cols. 27-35 (F-1)	Particulates	(micrograms/cubic meter)
Cols. 36-44 (F-2)	SO_2	(parts/million)
Cols. 45-53 (F-3)	CO	" "
Cols. 54-62 (F-4)	NO	" "
Cols. 63-71 (F-5)	Hydrocarbons	" "

No attempt will be made here to suggest realistic values. They may be obtained from the various APCO reference materials.

Over-ride Automatic Calculation of Industrialists' Dividents:

Normally the program will automatically calculate dividends to be deducted from each industrialist's cash balance. The amount deducted is usually all of the cash increase over 00,00. If the game director chooses to override the automatic calculation of dividends (for example, if he has some prior agreement with the industrialist or knows that the industrialist has applied for a loan to increase cash on hand for the purpose of expansion but has not yet spent it) he may do so by means of the DV card.

Cols. 1-2	DV
Cols. 3-6	Industrialist number (either 1-6 or 8-13 is acceptable)
Cols. 27-35 (F-1)	Dollar amount of dividends to be deducted.

Change Industrialist Average Annual Wage:

The game director may stage some labor strife for one or more of the industrialists which results in a change in average wage scale for his employees. If so, the new wage scale should be punched in the F-4 field (cols. 54-62) of the PL card for that industrialist. Note: this field is not included on the industrialists worksheets or coding forms since it is not a normal player option. However, if some industrialist feels particularly altruistic he may request a greater increase than normally produced automatically by the program.

Change State Equalization Factor for Property Tax Calculation:

This is a very rarely used option but one which may be employed by the game director if he wishes to announce that the State Board of PropertyEqualization has discovered that a particular jurisdiction is not assessing property at the legally required 50% rate. The equalization factor adjusts the actual assessment percentage to 50% before applying property tax millage. Note: only jurisdictions 1 through 4 can be changed. County millage is applied against the taxable property value of each jurisdiction. The method of entering this value is:

Cols. 1-2	AC
Cols. 3-6	Jurisdiction (only 1 through 4 may be changed)
Cols. 27-35	Assessment factor. This must be entered as a proportion, i.e. a fraction less than 1.0.

Cash Transfers To or From Operator To or From Any Player:

This is a very convenient mechanism for either adding cash to or removing cash from any player's account without effecting the account of any other player. It can be used for correcting unintentional depletions of a player's funds as, for example, restoring an over deduction of dividends for an industrialist. It could also be used to give federal or state subsidies to developers for low-income housing development or used to fine developers or industrialists for zoning infractions. The method is the use of a Cash Transfer (CT) card, the same as that used for cash transfers between game players, except that a code of 99 is used in either the "to" or "from" field to indicate that the destination or source is "operator".

Special Grants to Politicians:

This option is similar to the Cash Transfer except that it is a method of adding cash only to one of the five political jurisdictions (usually only Central City or County) and the Special Grant can be earmarked for either the General Operating Fund or the Capital Improvement Fund. It is shown on the Player Output for the Politicians as a specific line item in both the General Fund and Capital Fund budget pages.

As many individual special grant cards as desired may be input to the program but it is something of a waste of effort since the program simply totals all special grants in the same category and prints only the total on the output. The method of input is:

Cols. 1-2	SG	
Cols. 3-6	Jurisdiction to which grant is made;	1 = Central City 5 = County
Cols. 7-10	Fund to which grant is made;	1 = Ceneral Fund 2 = Capital Improvement Fund
Cols. 27-35	Dollar amount of grants to that fund.	

Federal Grants for Air Pollution Control Program:

This is an important part, usually the major part, of the Air Pollution Control Office budget. These funds are allocated by the Game Director or someone he has designated as the federal representative in response to applications by the Air Pollution Control Officer. They may be either new grants or supplementary grants and may be for one, two, or three years. These funds are input on a special form of the APCO's Budget Request (BR) card, usually referred to as the BR-3 card.

Cols. 1-2	BR
Cols. 3-6	0003
Cols. 27-35	Total dollar amount granted for current cycle. This is the total, not for any one of the six lines in the APCO budget. If the APCO already has Federal Funds allocated for this cycle from a previous cycle, this becomes a supplementary allocation and is added to the amount already in the APCO file.
Cols. 36-44	Federal funds (dollars) for next cycle. The above comments apply.
Cols. 45-53	Federal Funds (dollars) for cycle after next.

Reducing Federal Air Pollution Control Allocation

The Game Director (or federal representative) has one other option for controlling the APCO's funds. Each cycle the APCO output lists federal funds granted for each of the following two cycles (a result of a two or three year grant on a BR-3 card above) and the APCO may make his budget for the next cycle on the basis of the federal funds he assumes he will be receiving. However, the game assumes that the government can change its mind. The Game Director may decide that he will introduce a "federal money squeeze" and has the option of specifying that some percentage less than 100% of the federal funds listed in the APCO output will actually be available. The operator does this by coding in the I-2 field (cols. 7-10) of the BR-2 card (the card on which the APCO allocates his federal funds to his six budget items) an integer percentage less than 100 which is the percentage of federal funds which will actually be available. Note: this value must be greater than zero. (A zero or blank field defaults to 100.) The program multiplies each of the six federal budget items by this amount and, if necessary, cuts the corresponding County budget items back to 1/3 of the federal budget.

Chapter 4

Error Messages

During the computer processing component of M.E.T.R.O.-APEX, many opportunities exist for error. Frankly, it is a rare occasion when the processing of a set of decisions is free from error. The need for quick and accurate corrective action is most dramatically demonstrated during runs requiring rapid turn around because processing errors disrupt player strategies (ommission of a single card might result in the non-execution of a crucial activity in connection with some major plan) and upset the morale of the audience and the timing of the game.

Errors arise from many sources. Players make errors in specifying the exact nature of their decisions. Coders of key punch forms make errors in transposition. Errors can be made by key punchers and, finally, the operator can communicate incorrectly with the program. The most difficult errors to correct are player mistakes--unless the role advisor is sufficiently close to the players to place the logic of player decisions correctly on the worksheets, the computer operator can not correct these mistakes. (The MISMATCH OF SCALE error is the most common error of this type.) Other errors can and should be corrected; unfortunately however, the addition of a zero on a number, the minor miscoding of a single number, etc., often go by undetected. One useful activity that often minimizes the occurence of such errors is to unconditionally halt after reading the Main Player Decision cards and visually check the listing for errors prior to advancing the program beyond that section. A second activity, if time is available, is the reading of the output for glaring inconsist-

encies and errors and the rerunning of the cycle to correct the mistakes. As the operator gains experience with the game, his speed and ability to catch errors before output is returned to the players should go up markedly.

This appendix takes up all the diagnostic error messages that may be printed during the normal run of M.E.T.R.O.-APEX. Each type of error is considered in a separate section of this appendix.

VERY IMPORTANT WARNING! Methods of restarting after error termination.

First, some background information:

After reading the Main Header Card, the M.E.T.R.O.-APEX program makes a decision about which files it needs to continue processing. After a normal termination of a cycle of the game, data set 15 contains a record of the final state of the variable player accounts for the cycle just concluded and data set 16 contains the status of the same accounts at the beginning of that cycle. In particular, at the end of the computer run for cycle 2, for example, the cycle number stored in data set 15 is 2 and the cycle number in data set 16 is 1.

The decision the program makes at the beginning of a run is as follows:

a. If CYCLE=n (n greater than 1) is specified on the Main Header Card, implying that you are going on to a new cycle, data set 15 is copied into data set 16 so that at the beginning of decision processing both data sets contain exactly the same information. When this copy operation has been successfully concluded the program prints the following message:

```
STATUS AT END OF CYCLE n HAS BEEN COPIED FROM DATASET 15 TO DATASET 16
FOR START OF NEW CYCLE
```

Once this message has been printed you must use REPEAT= on the Main Header Card to restart the cycle due to any errors which may occur after it (as well as to simply rerun the cycle with different decisions). If fatal errors have occurred prior to printing this message (errors in JCL or on the Main Header Card itself) you should simply correct the error(s) and restart retaining the CYCLE= on the

Main Header Card.

b. If you had specified REPEAT=n (n greater than 1) on the Main Header Card, the program copies data set 16 into data set 15. When this copy operation has been successfully completed the program prints this message:

DATASET 16 HAS BEEN COPIED TO DATASET 15 FOR REPEAT OF CYCLE n

Since the records in data set 16 are not changed during processing you can restart using REPEAT=n for any errors which may occur.

c. CYCLE=1, REPEAT=1 and NEWGAME are synonymous. They each result in data set 14 (the base team data, with cycle number 0) being copied into both data set 15 and data set 16. (This is the only time data set 14 is used.) Since this is the case, any errors in running cycle 1 can be corrected and restarted using any of the above options.

Notice that the above restart procedures apply only to errors which may occur in input card sets 1 through 6. Assuming the first Main Section of the program runs correctly, any errors in input card sets 7 through 10 (issue and newspaper data) should be corrected and restarted using the NEWSPAPER option on the Main Header Card and removing input card sets 3-6 from the input deck. (See Chapter 3.)

A. Errors in Job Control Cards:

Errors which may occur in the Job Control Language at the beginning of your input deck could be many and varied and will not be dealt with at length in this volume. Any that do occur will probably be due to discrepancies between the requirements of your installation and the sample set of JCL shown in Chapter 3, Figure 3. Among the most probable sources of error are:

Punching the incorrect VOL=SER=code on the DD cards, particularly FT15F001 and FT16F001 if running multiple teams or saving all cycles.

Incorrect UNIT or DCB parameters on cards FT03F001 and FT09F001.

You should consult with a representative of your computer installation for

diagnosis of JCL errors.

B. Errors on the Main Header Card:

As each program control card is read by the computer it is printed on the output for record purposes whether or not it contains any error. The program section which scans and interprets the Main Header Card will print a $ sign under the line on which the Main Header Card is printed in the position where an error is detected. The type of error will be indicated on the following line. All possible errors will be checked on the Main Header Card and up to 10 errors in Additional Program Control Cards will be detected before the program will terminate.

*** ILLEGAL OR CONTRADICTORY KEYWORD ON HEADER -- '____'

Only the first four characters of the Main Header Card keywords are checked. The above message will be printed if the program does not find one of the following: NEWG, CYCL, REPE, NEWS, TEAM, DATE. The four characters which are considered illegal are printed following the message. One possible source of illegal keywords could be the use of apostrophes (') within the team identification field. This string of characters starts with an apostrophe and is considered ended when the next apostrophe is detected. Therefore any additional characters following the second apostrophe is considered another keyword.

A keyword is considered contradictory if it appears more than once on the Main Header Card or, with one exception, if more than one of the keywords NEWGAME, CYCLE, REPEAT or NEWSPAPER is used. The one exception is that the keyword CYCLE may follow the keyword NEWSPAPER.

*** ILLEGAL NUMBER -- '____'

The following are considered illegal numeric values:

1. Any value less than 1 or greater than 10 following CYCLE or REPEAT, except
2. Any value other than 0 following the combination NEWSPAPER CYCLE.
3. Any value less than 1 or greater than 5 following TEAM.

4. Any negative or zero value in any of the three DATE fields. (Unfortunately we will either have to modify the program or not play M.E.T.R.O.-APEX in the year 2000.)

*** DID NOT FIND 'CYCLE', 'REPEAT', OR 'NEWGAME' ON HEADER CARD

This message is printed if one of the keywords NEWG, CYCL, REPE or NEWS is not on the Main Header Card.

*** TEAM NUMBER NOT GIVEN ON HEADER CARD

Self-explanatory. The team number is required.

*** DATE NOT GIVEN, WILL USE DATE FROM PRECEDING CYCLE mm/dd/yy

The above is not a fatal error, but only a warning. If the date is not specified, the date saved in bytes 3-8 of "file 16" will be used. If starting a newgame this date will be all zeros and will be found incorrect at a later point, therefore DATE should always be specified when starting a new game.

*** ILLEGAL CHARACTER IN EXPECTED NUMERIC FIELD

Only the ten numeric characters, 0 through 9, blanks (which are ignored), or the terminating character comma (or / for date fields) are legal following CYCLE, REPEAT, TEAM or DATE. Any other character will give the above message. The most usual causes are (1) failing to terminate the number with a comma or (2) failing to separate the number from the preceding keyword with a blank or equal sign. For example, both of the following would produce the above error message:

```
CYCLE3, TEAM 1
CYCLE 3 TEAM 1
$
```

*** SLASH ILLEGAL HERE

Slashs are only legal separators between parts of the DATE field. Any other occurrance following a number will produce the above error.

*** mm/dd/yy IS AN IMPOSSIBLE DATE

The program contains an algorithm which checks each field of the date for correctness. The month must be 1 through 12, the day must be greater than zero and less than or equal to the correct number of days in the corresponding month (a check for leap-years is included), and the year must not be equal to zero (as noted above).

**** TEAM NUMBER/FILE MIXUP

TEAM NUMBER IN DATASET SPECIFIED ON FT[15 16]F001 IS n

TEAM NUMBER ON HEADER CARD IS m

The team number (from 1 to 5) is stored in the bytes 75-76 of record 71 of data sets 15 and 16 by utility program NEWTEAM (see section on Utility Programs for M.E.T.R.O.-APEX) which must be run prior to the first time a game is run for any team. Each time NEWGAME, CYCLE 1 or REPEAT n is specified the team number in both files is compared with the team number on the Main Header Card. When any CYCLE other than 1 is specified the team number in data set 15 is compared with the team number on the Main Header card. The above message is printed if the numbers are not the same to prevent inadvertant destruction of the wrong team files.

The error is caused by either (1) not having preset files for a new team with the NEWTEAM utility or (2) either the wrong data set DSNAME specified on the appropriate DD card or the wrong team number on the Main Header Card.

*** CYCLE NUMBER IN DATASET 16 WAS n. SHOULD HAVE BEEN m.

CHECK NUMBER OF REPEAT CYCLE OR DSNAME ON FT16F001 CARD.

This message will only occur in conjunction with REPEAT on the Main Header Card. The cycle number in bytes 1 and 2 of record 71 of data set 16 must be one less than the cycle being repeated; in other words, the record at the end of the cycle preceding the one being repeated. If you are using the Basic Job Control in Chapter 3, this error could be caused by trying to repeat a cycle earlier than the last cycle completed. If running multiple teams or saving the records

of all cycles it could be caused by the wrong DSNAME on the FT16F001 DD card.

*** CYCLE NUMBER IN DATASET 15 WAS n. SHOULD HAVE BEEN m.

CHECK CYCLE NUMBER ON HEADER CARD OR DSNAME ON FT15F001 CARD.

This message will only occur in conjunction with CYCLE on the Main Header Card if the cycle is greater than 1. It could be caused by attempting to run a new cycle when the preceding cycle had not run to completion or by attempting to repeat or start back at an earlier cycle without specifying REPEAT on the Main Header Card. The cycle number in bytes 1-2 of record 71 of data set 15 must be one less than the Cycle number specified on the Main Header Card. (See section on the Logic of File Saving.)

C. Errors in Additional Program Control Cards:

Each Additional Program Control Card (input card set #3) is printed on the output for record purposes as it is read. If any errors are detected on any of these cards, an appropriate message will be printed immediately following the card image. A maximum of 10 errors will be detected by the section of the program which reads the Additional Program Control Cards. If more than 10 errors are detected the program will print:

TOO MANY ERRORS. EXECUTION HALTED.

and the program execution will be terminated.

The following errors may be detected:

xxxx MAY NOT BE SUBSCRIPTED.

The parameter specified as xxxx has been subscripted and should not be; e.g. NOPLAY=ALLDEV(1) is incorrect.

xxxx MUST BE SUBSCRIPTED.

The parameters DEVELOPER, INDUSTRIALIST, POLITICIAN, AND PLANNER or their abbreviations must be subscripted.

xxxx HAS AN OVERSIZE SUBSCRIPT.

The maximum subscript value for DEVELOPER and INDUSTRIALIST is (7) and for

POLITICIAN and PLANNER the maximum subscript is (5).

xxxx IS UNDEFINED.

The keyword or parameter xxxx is not a valid word. Probably misspelled or incorrectly abbreviated word. Check the list in Chapter 2.

UNEXPECTED END OF FILE or

UNEXPECTED EOF

These messages should only be printed if the program detects and "end-of-file" in the input deck before it finds a RUN or DATA card. An end-of-file card for the input deck is the card with /* in the first two columns or the end of the input card deck. It might occur, for example, if you were running exact copies of the newspaper and had omitted the RUN card between the Main Header Card and the Newspaper Header Card.

Normally the M.E.T.R.O.-APEX program terminates with one of the two following messages:

*** NORMAL METRO-APEX JOB TERMINATION

if no errors were detected in the entire program or

*** METRO-APEX ABNORMAL TERMINATION WITH nn ERRORS

Any occurrance of the UNEXPECTED EOF message (and ABEND 0077) after either of the above messages should be disregarded.

D. Errors in Reading the Main Player Decision Cards:

As the Main Player Decision Cards are read by the program error diagnostics are written on the printer under the line containing the image of the incorrect card. If there are any errors, or if you have submitted the job only to get a listing of the input data deck and have found errors not detected by the program, the errors must be correct and the job resubmitted specifying REPEAT on the Main Header Card.

Each error message is described briefly in the following paragraphs.

Several errors are associated with project and program input cards.

*** ILLEGAL PROJECT NUMBER [n] ***

If the number entered on a project or program card is not associated with any existing project or program, the above message is printed.

*** MISMATCH OF SCALE . . [n] ***

Assignment of a project or program to a location which is larger (more inclusive) than the scale indicated on the project list will cause this error message to appear; for example, coding 30 (Ward 1) for a project which must be assigned to a specific analysis area.

*** MISASSIGNMENT OF AREA [n] ***

When analysis areas are coded incorrectly, most often when projects restricted to a particular area are assigned to another area, this message is printed.

*** PRICE (F-1) OUT OF RANGE ***

When prices on capital improvement projects are outside the range specified on the project lists, this error is printed.
Another message may appear after a capital improvement or program card.

*** FIELD I2 (LOC) CHANGED TO [n] ***

This message is not an error. Assignment of a project or program to a location which is smaller (less inclusive) than the scale indicated on the project list will activate a mechanism within the program to change that location automatically to the proper larger scale and write out this message.

Errors in the balance of the input cards will trigger the following diagnostics.

*** NUMERICAL VALUE [n] IMPROPER ***

An entry on a card that is clearly impossible (for example, a zero entry for a player number) will trigger this message. The number [n] indicates which of the 12 possible numeric fields is in error.

*** NON-GAME LD OR IND ***

If a developer or industrialist number appears on a card but is the number of a non-playing developer or industrialist due to inclusion on a NOPLAY card, the above error is printed.

*** ALPHA CODE UNKNOWN ***

If the computer can not interpret the first two letters on a card, the above message is printed.

*** BR/SR COMBO WRONG ***

If, on land transfer cards, the buyer and seller are the same, the above message is printed.

*** INCORRECT BOND/MILL NO. ***

This message is written in one of two cases. If the card requesting a bond or millage issue (BS card) is not read prior to the project or program request (CP or SP card) tied to that bond or mill or a campaign contribution to that bond/mill, this message will appear. Rearranging the input to place the BS card before the CP, SP or CC card(s) will rectify this error. The error is also caused if the first digit of the bond/mill number is not the same as the cycle number; i.e. in cycle 2 the bond/mill numbers must be 201, 202, 203, etc.

*** MORE THAN 20 BONDS/MILLS, ABOVE CARD IGNORED ***

If more than 20 requests for Bonds or Mills are processed in a cycle, the above message is printed. It may be corrected by either omitted the card or cards over 20 (not a very desirable solution) or, preferably, by combining one or more bonds or mills of the same type for the same jurisdiction into a single larger bond issue or millage. If you do make corrections of either type, don't forget to check the CP, SP, and CC cards for references to the BS cards which were eliminated.

One more error message is printed in response to problems on cards dealing with land transactions.

*** COMBO IMPROPER ***

F. Errors in Reading Issue Decisions:

A subroutine, DECOD, is used to scan and interpret issue decisions (and CEM input cards) of the form iiia (bbb). DECOD searches for Syntax errors only. The only legal characters which may appear on an issue decision card are:

blank 1 2 3 4 5 6 7 8 9 0 ( ) , *

If a character other than the above is found, this message is printed.

ILLEGAL PUNCH IN COL [n]

This message indicates the column containing the invalid character to assist the operator in making the needed corrections. Since the maximum number of digits in a single string on a card is four and the minimum is one, a check is made for excessively long and short combinations. Strings of unacceptable length result in either of two messages.

MORE THAN 4 DIGIT NUMBER, COL [n]

LESS THAN 1 DIGIT NUMBER, COL [n]

If only one digit is found between two commas, it is assumed to be an alternative number with no preceding issue number and results in the following.

ZERO ISSUE NUMBER

Note, that no column number is given--the operator must scan the incorrect card for the error.

Whenever a right parenthesis ")" is encountered, the program checks that there was a preceding left parenthesis. The failure to pass such a check results in the following.

WRONG PARENTHESIS, COL [n]

The entire set of issue decisions on a single card is checked for errors before execution is terminated. After scanning the entire card, if any errors were found, the following message is printed:

***** nn ERROR(S) ON ABOVE CARD *****

Illegal combinations of use and zoning. For example, if use is vacant (code 0) the zoning must be 1-6. If developed (code 1), zoning must be 1-11.

One last message may appear on the output as a warning of a potential error.

*** MISSING APCO COUNTY BUDGET CARD ***

This is likely to be a serious error unless you are not playing the APCO role, since without a county budget allocation the APCO will have no money.

The messages listed above are responses to errors in input found by the M.E.T.R.O.-APEX program itself--the program checks the cards for mistakes and inconsistencies and reports out any errors. Sometimes, however, errors are made in preparing the input cards of a nature that prevents the computer from correctly reading the cards. these errors cause the computer itself to print diagnostic messages and take some action which depends on the Operating System generated at your installation. Probably the only error of this type which you are likely to encounter is the following:

IHC215I CONVERT - ILLEGAL DECIMAL CHARACTER X

This error results from having punched some character other than a decimal digit 0 through 9k a single plus or minus sign or blanks in an integer (I) field or the above characters plus a single decimal point in a real (F) field.

The action taken depends on whether or not the Operating System at your installation was generated specifying what is known as the "extended error handling facility." If extended error handling is not in effect the system will halt when the above error is encountered. You must correct the error and re-start specifying REPEAT on the Main Header Card. If the system does have the extended error handling facility the problem is compounded. The system pints the diagnostic, performs what is known as the "standard fixup", which consists

of setting the value of the field in error to zero, and continues execution. Since these errors are detected by the system prior to the checks by the M.E.T.R.O.-APEX program it is possible that the zero field will be detected as an error by the main program and result in an error termination after reading all of the data cards. It is also possible, however, that the zero field will not be detected by the M.E.T.R.O.-APEX program during card input but that execution will continue with the incorrect value. If this does occur you should check the entire output for possible errors as a result of this zero field and, if necessary, REPEAT the cycle, printing only the section of the output effected.

E. Errors in Reading Newspaper Header Card:

The Newspaper Header Card is scanned for legal keywords in the same manner as the Main Header Card (Section B.). Only the first four characters of each keyword are considered. The legal combinations are: COPI, COPY, IGNO, RERE, NOHE, CEMI, NEWI. (See Chapter 3 for complete keywords.) If some four-letter combination other than the above is detected, the incorrect column is flagged and the following message printed:

*** NOT A LEGAL KEYWORD ON NEWSPAPER HEADER CARD

If a number followed by a comma is not detected following COPY= or COPIES= (probably due to a missing delimiter between the keyword and the number) the following message is printed:

*** ILLEGAL CHARACTER IN NUMERIC FIELD. COPIES SET EQUAL TO ONE.

Note that the number of copies is not required on the Newspaper Header Card. If not specified the number of copies is set by default to one.

If any errors are detected on the Newspaper Header Card the following message is printed and execution terminated:

*** EXECUTION TERMINATED DUE TO ABOVE ERRORS. CORRECT AND RERUN
SPECIFYING 'NEWSPAPER' ON MAIN HEADER

If there are no syntax errors on the Issue Decision Card(s), the program checks the content of the cards. Two warning messages may be printed.

MORE THAN 40 SETS OF NUMBERS. STOPPED AT 40.

The maximum number of issues that can be stored at a single time is 40. Additional issues are ignored. The above message indicates that such "ignorance" is present.

WARNING -- NO TERMINATING CHAR. LAST NUMBER NOT STORED

The last character on each issue card must be either an asterisk or a comma. Any characters appearing on a card after the last comma (but not followed by an asterisk) will not be carried over to the continuation card but will trigger the message printed above. This is only a warning inas much as the incomplete issue entry triggering the message may be restarted on the continuation card without deleting the partial entry from the first card. If the incomplete entry is not repeated, it is, in fact, an error.

In the absence of warnings and/or syntax error messages, a series of messages may be printed indicating that the entry for a given issue is invalid. If the issue number is not within the 1-300 range, does not correspond to an initialized entry in the data file, or has an incorrect alternative number associated with it, one of the following messages is printed.

ILLEGAL ISSUE NO. [nnn]

or

ILLEGAL ALTERNATIVE (n) FOR ISSUE [nnn]

If any of the above messages are printed the program halts after printing:

CORRECT ERRORS AND RERUN SPECIFYING 'NEWSPAPER' ON MAIN HEADER CARD

If bond or millage numbers are entered on the issue cards, the program compares the first digit of the bond or mill number to the cycle number. If they are not the same the following is printed:

ILLEGAL BOND NO. (nnn) FOR ISSUE m
IGNORED IN SUBSEQUENT PROCESSING

Note that the above does not terminate processing but is a warning only.

After all of the above checks, if no errors are detected, the program compares the list of issues read from the issue decision card with the list of issues from the preceding newspaper which had possible linkages to other issues. If the program does not find a decision for those issues the following is printed:

*MISSING DECISION FOR ISSUE n

for each such missing decision. If you had specified IGNORE on the Newspaper Header Card processing will continue. If you had not specified IGNORE, the program will terminate after printing:

*** "IGNORE" NOT SPECIFIED. PROCESSING TERMINATED DUE TO MISSING DECISION(S).

One last message may appear after reading the issue decision cards. The program saves the record of the decisions from cycle to cycle for use in processing the candidate election model. A maximum of 40 issues can be saved from election cycle to election cycle. If that number is exceeded, the following is typed.

MORE THAN 40 ISSUES FOR CEM, nn IS IGNORED

This error is not disastrous. Because it is possible to input a special set of issue decisions for the Candidate Election Model, the issue list can be edited (to eliminate repetitions) when that model is executed.

G. Errors in Reading Candidate Election Model Input:

The input stream recording the candidate decisions on issues is scanned by the same DECOD routine that interprets issue decision cards. Therefore, any or all of the syntax errors described in connection with the issue decision cards may appear in response to errors in the candidate decision cards.

In addition, a series of straightforward error messages may be printed.

ILLEGAL PLAYER NUMBER nn

This is printed if the player number is less than 1 or greater than 16.

ILLEGAL OPPONENT TYPE FOR CANDIDATE nn

This is printed if the opponent type for incumbent cards (player number 1-8) is not 1 or 2. It also appears if the simulated opponent type on the opponent cards (player number 9-16) is not a digit between 1 and 9.

DUPLICATE PLAYER NUMBER nnn

This message signals that more than one set of cards was entered for player nnn.

NO MATCHING CANDIDATE FOR CAND. nnn

For every incumbent card read (candidate number 1-8) a corresponding opponent card (candidate number 9-16 with 1 paired to 9, 2 to 10, etc.) must be read. A mismatch in pairing triggers this last message.

If any of the above error messages are printed, the program terminates and the errors must be corrected and the program resubmitted, specifying NEWSPAPER on the Main Header Card.

In addition to the above "fatal" errors, the following errors may be detected. They are not considered "fatal" to execution because the indicated corrective action is taken, but they should be analyzed carefully for their effect on the Candidate Election and, if necessary, the cards should be corrected and the NEWSPAPER rerun.

PLAYER nn SPENT MORE THAN HE HAD (nnn PCT), CHANGED TO

FF.fff FF.fff . . .

The designated player entered percentages totalling more than 100% in assigning his funds to stress the various issues. The program automatically adjusts this "oversight" by cutting the percentages back proportionally.

WRONG PARTY SUPPORT FOR CAND. nn. SET=30.0

This message appears if the value in columns 4-8 of the incumbent card is neg-

ative or greater than 60.0. (Warning only.)

PERCENT OVER 100 FOR ISSUE nnn CAND. nn. SET=100

This is printed if the percentage appearing in parenthesis for a given issue (designating stress) is greater than 100. (Warning only.)

ISSUE nnn FOR CANDIDATE nn NOT IN MASTER LIST -- IGNORED

The program verifies that all issues entered for a given candidate were included on a previous issue decision card or input specially. If no record of an issue is found, the above diagnostic is printed. (Warning only.)

Chapter 5

The FILEPRNT Utility Program and Other Options

The FILEPRNT Program--Creating and Printing M.E.T.R.O.-APEX Files:

This utility program controls, by selective subroutine calls, all the necessary functions of initializing the "files" in data sets 14, 17, 18 and 19. Normally all of the required data cards for creating all of the files are in a single input stream on magnetic tape with the appropriate control cards for this program separating the individual decks. This initial input stream also causes a printout of all of the files (except issues and headlines) as they are created.

Once the files are created, this program is used to print any of the files, including those in data sets 15 and 16, either for diagnosis of program errors or for operator or player information. Of particular importance, since a number of copies will be needed for each game run, is the ability to print lists of capital projects and special programs (File 6). It is also useful for the operator to have a list of preset issues and headlines for each cycle and a complete listing of all issues and headlines (ISLST).

The program is executed using the same Job Control card set as indicated for the main METAPEX program in Figure 3 in Chapter 2 with the following exceptions:

1. Replace the word METAPEX on the EXEC card with the word FILEPRNT.
2. (Optional) The FT09F001 DD card is not necessary.
3. The REGION= size on the JOB card can probably be eliminated, depending on the default region size at your installation. The FILEPRNT

will execute in a region size of 80K for file printing. A larger region size will be required for initial creation from magnetic tape if the block size is 10,000 bytes. (This information will be supplied when the program is delivered to your installation.)

The control cards for the FILEPRNT program are simple, involving a single keyword starting in column 1 on each card, in some cases followed by one or more decimal numbers. The following are the legal keywords and their meaning:

MAKE n -- this card indicates that the program is to make file n. This card is immediately followed by the data cards to be read by the program to make file n, if required. (Files 35 and 36 require no data cards.) The only legal values for n are: 3, 5, 6, 21, 28, 30, 31, 32, 34, 35, 36, 37, 38, 39, 53, 61, 62, 64, 68, 201, 203, 204, 206, 209, 211, 237. (Files 5 and 209 are the same, the values are synonymous.)

Note: the file creation routines for files 6 and 28 also require control cards in addition to those required by the main program. They will be described in a following section.

The formats for data cards required by the above file creation routines will be described in a supplementary document.

PRINT n,m -- print the contents of file n. The legal values for n are:

1, 31, 41
4, 34, 44
7, 37, 47
8, 38, 48
15, 35, 45
16, 36, 46
19, 39, 49
20, 30, 40
3, 6, 11, 12, 21, 28, 53, 513, 204, 206, 211, 201*

*PRINT 201 is the same as using the keyword ISLST. Both print the contents of files 201, 202 and 203--the so-called issue list.

22, 32, 42

51, 61, 71 — File 6 is the only file that is usually printed

52, 62, 72 — with any regularity. The format of most of the

54, 64, 74 — other file prints is usually meaningful only to

58, 68, 78 — programmers.

5, 209

511, 531, 541

512, 532, 542

Files 1, 31, 41 and 11 may be followed by at least one additional value, separated from n by a comma. This value designates the analysis area to be printed for files 1, 31 and 41 and the record from 1 to 50 to be printed for file 11. The reason is that a complete dump of either of these files is very lengthy and usually of little value. If m=99 for any of these files the complete file will be printed. For example: PRINT 1,23 would print only file 1 for analysis area 23.

As many PRINT cards as desired may be included in the input stream.

ISLST n,m -- print the issues, alternatives and headlines in files 201 and 202.

The parameters n and m are optional. If omitted the entire list of issues will be printed. If specified n is the first issue to be printed and m is the last. For example:

ISLST 1,10 will result in printing only the first ten issues.

ISLST 101 Would print all of the issues from 101 to the end.

Note: the issue file (#201) contains a maximum of 300 issues. At the present time only about 180 are used, allowing for future additions. Only those actually in the file will be printed. There is room in the headline file (#203) for 1498 headlines. Many of these headlines are printed as part of issues. When all issue related head-

lines have been printed, the program will continue with the balance of the "one-liners" which have not already been printed as part of issues or alternatives. At the end, the program will list the numbers of issue records and headline records which have not yet been used and are available for expansion.

BKVIN -- this calls a special routine which does a final initialization of files 30 and 34. There are no data cards for this routine. It must be run following the original creation of files 30, 31, 32, 34 and 64 from cards. All of the mentioned files must already have been created when BKVIN is run. It should not be used except in very special cases.

NOPRINT -- the main FILEPRNT program is set to automatically call the print routine corresponding to a particular file following the use of a MAKE control. There may be circumstances when you may not wish to get a complete dump of all files created. The NOPRINT keyword card turns off this automatic print feature. It may be used preceding any MAKE card and remains in effect until:

PRON -- this keyword, an abbreviation for PRintON, turns automatic printing back on following the use of NOPRINT.

FINISH -- as many of the above control cards as desired may be included in a single run of FILEPRNT. The last card in the input deck must be FINISH to indicate the end of the input deck to the program.

Control Cards for Creating and/or Printing File 6--Capital Projects:

A special section of the FILEPRNT program facilitates the addition of or changes to the file of capital projects and special programs. It also allows deletion of projects and/or programs when they are no longer needed. Once added, a new project or program should remain in the file for the balance of the game; however, all new projects should be deleted from the file prior to the initiation of a new game run.

Note that since the FILEPRNT program is entirely separate from the main METAPEX program it must therefore be executed before running METAPEX for any cycle in which there is to be a reference to a new project or program. The input to this special program is made from cards punched with the following format.

Project Card*

Cols.	Format	Item and Code
1-3	I3	Three digit integer project number (1-120)
4-5	I2	Location: 01-29 for Specific Analysis Area (A.A.) 30 for any A.A., 31 for any Ward 32 for any Jurisdiction
6	I1	Type: 1 = Streets, 2 = Sewers, 3 = Water, 4 = Parks and Recreation, 5 = Miscellaneous
7	I1	Capital Plant Index Impact: 0 = A.A., 1 = Ward, 2 = Jurisdiction, 3 = County
8	I1	Years to Construct
9	I1	0 if Revenue Bond Financing is not allowed, 1 if it is.
10-14	F5.0	Acres required (Real number with decimal)
15-23	F9.0	Total Minimum Cost (Real number with decimal)
24-32	F9.0	Total Maximum Cost (Real number with decimal)
33-77	45A1	Project Title

Program Card*

Cols.	Format	Item and Code
1-3	I3	Three digit integer program number (1-40)
4-5	I2	Number of years program is to run
6-7	I2	Location: 1-29 for a Specific Analysis Area (A.A.) 30 for any A.A., 31 for any Ward 32 for any Jurisdiction and 33 for County-Wide
8-10	I3	Required project number (leave blank if none is required)
11-20	F10.0	Cost per year
21-68	48A1	48-Character Title
69-70	2X	Blank
71-73		Number of issue to be printed during the last year of the program (usually blank for additions)

All integer fields (I Formats) must be right justified.

The control of the program for inserting or deleting or printing projects is by means of keyword control cards which are in addition to those required by the main section of the FILEPRNT program. They are:

INITIAL -- this should only be used when you wish to completely wipe out all projects and programs in the file and completely reload the file. It does not result in reading any cards, it simply clears the file. It must be followed by additional control cards to read new data.

PROJECTS -- this indicates that the cards which follow are capital project cards punched according to the preceding format. The last card in the deck of project cards must be blank to indicate the end of the project card deck.

PROGRAMS -- this control card indicates that the cards which follow it are special program cards punched according to the preceding format. The last card in the special program deck must be blank.

DELETE n,m,i, ... n -- this card contains a list of projects and/or programs which you wish to delete from the file. Each number must be separated from the next by a comma. Note that special programs must be number from 121 to 160.

REPLACE -- normally when adding projects and/or programs the program will indicate an error if a record with that number already exists in the file, assuming that you do not wish to destroy a record that already exists. The above control card may be used preceding a PROJECT or PROGRAM control card to allow replacement of records which already exist.

LIST n -- this control card will result in the printing of n copies of the project/program list. The default if n is not specified is one.

STOP -- this card must be used to indicate the end of all control cards related to file 6 and causes a return to the main FILEPRNT program.

Running More Than One Game:

Some organizations using M.E.T.R.O._APEX may need to be able to process more than one run of the game at the same computer installation concurrently. The reasons for this can be many -- short demonstration runs overlapping long-term runs, more than one course using the game during the same semester,

training sessions for operating staffs each of which is advising a different team, or running for very large groups where the group is divided into two or more teams.

The IBM 360 computer program for M.E.T.R.O.-APEX can be used to run as many teams as necessary, although we recommend that no more than five game runs be in progress at any one time since the file handling becomes rather cumbersome for more than five teams and can lead to unintentional errors. The procedure for up to five teams is extremely simple, since it only involves keeping two additional data files for each extra team.

Chapter 6, M.E.T.R.O.-APEX Data File Organization and Description, has discussed in detail the six "permanent" data sets used by the M.E.T.R.O.-APEX/360 program. It was noted that data sets 14, 17 and 19 contain files which are "read-only" within the main program (the data may be changed for special purposes by the FILEPRNT program) and are shared by all teams. Data set 18 contains variable records of issue and newspaper decisions, however it is set up to handle records for up to five game runs automatically and no modification to the standard Job Control set is required for running from one to five teams.

Data sets 15 and 16 are the critical data sets from the standpoint of multiple teams. Each team must have its own pair of data sets 15 and 16. Before running the game for any team for the first time, data sets 15 and 16 must have been "preformatted" by running a special job which creates the two data sets by reserving space on a disk volume for them, names them and runs a program which writes the team number into a specific place in each of the two data sets (bytes 75-76 of record 71). An example of this job is shown in Figure 8 . When the main program is executed, the team number specified on the Main Header Card is compared with the team number stored in one or both of these data sets and, as noted in the chapter on Error Messages, execution will

Figure 8

JOB TO PREFORMAT TEAM FILES

```
//NEWTEAM  JOB  (9224120,APEX),PRATT,CLASS=E,TIME=1
//STEP1    EXEC  FORTGCLG
//FORT.SYSIN  DD  *
C
C     MAIN PROGRAM TO PREFORMAT NEW TEAM FILES
C        PLACES TEAM NUMBER IN PROPER LOCATION IN DATA SETS 15 AND 16
C
C          USE ONLY FOR CREATING NEW DATA SETS FOR ADDITIONAL TEAMS
C          ONCE CREATED THE TEAM NUMBER WILL REMAIN UNTIL THE DATA SET
C          IS SCRATCHED FROM THE DISK.
C
      IMPLICIT INTEGER*2 (I-N)
      DEFINE FILE 15(305,160,U,KX15)
      DEFINE FILE 16(225,160,U,KX16)
      DIMENSION A(160),NEED(40)
      EQUIVALENCE(A(1),NEED(1))
      DO 10 I=1,160
   10 A(I)=0.0
      DO 20 I=1,225
      WRITE(15'I) A
   20 WRITE(16'I) A
      DO 30 I=226,305
   30 WRITE(15'I) A
      READ(5,1) NTEAM
    1 FORMAT(I1)
      IF(NTEAM) 100,100,40
   40 IF(NTEAM-5) 50,50,100
   50 NEED(38)=NTEAM
      WRITE(15'71) NEED
      WRITE(16'71) NEED
      CALL EXIT
C
C       INVALID TEAM NUMBER
C
  100 WRITE(6,2) NTEAM
    2 FORMAT('0*****',I3,'  IS INVALID TEAM NUMBER')
      CALL EXIT
      END
/*
//GO.FT15F001  DD  UNIT=2314,DISP=(NEW,KEEP),SPACE=(640,(305)),
//         DCB=(RECFM=F,BLKSIZE=640),VOL=SER=[KIEL01],DSN=[FILE15.T[2]]
//GO.FT16F001  DD  UNIT=2314,DISP=(NEW,KEEP),SPACE=(640,(225)),
//         DCB=(RECFM=F,BLKSIZE=640),VOL=SER=[KIEL01],DSN=[FILE16.T[2]]
//GO.SYSIN DD *
[2]
/*
```

← Use appropriate JOB card for your installation.

Change to correct disk volume serial number at your installation.

← Punch team number in Col. 1 of this card. It should correspond with DSName on DD cards.

be terminated if the team numbers do not agree.

Data Set Numbers vs. Data Set Names:

A number of references have been made above and elsewhere in this volume to data sets by number; for example, data set 15, etc. This has been convenient for explanation but it is not strictly correct terminology. The following definitions will help to avoid confusion in the discussion which follows:

Data Set: An organized collection of related data residing on some storage medium; for the M.E.T.R.O.-APEX program it is normally stored on a direct access volume, usually a disk pack.

Data Set Name: Each data set is assigned a name at the time it is created and that name is used thereafter to refer to that particular data set. The name is specified to the Operating System on a DD card by means of the parameter DSNAME= or its abbreviation DSN=. Each data set on the same disk volume must have a unique name. An unqualified name consists of from 1 to 8 characters, the first of which must be an alphabetic character. A name may be qualified by appending to it a period followed by a qualifying name of from 1 to 8 characters, the first of which must be alphabetic. For example:

FILE15 is an unqualified name.

FILE15.TEAM1 is a qualified data set name.

FILE15.Team2 is a qualified data set name.

> are unique names.

Additional qualifications may be appended, subject to the same rules, up to a maximum of 44 characters. For example:

FILE15.TEAM1.CYCLE4 is a legal data set name.

Notice that there may be no embedded blanks in a data set name.

Data Set Reference Number: Within the computer program there is no reference to any specific named data set for reading or writing data. Instead the program refers to a data set reference number--it is this data set reference number (abbreviated DSRN) that is frequently used in this volume to refer

to data sets. The actual named data set to which any number refers is not known to the program until it attempts to read from or write into the data set. Then the Operating System supplies the program with the actual location of the data set which it obtains from the DD cards supplied to it as part of the Job Control Cards. The connection between the data set reference number in the program and the data set reference number on the DD card is coded within the DDNAME, that is, the FTnnF001 in columns 3-10 of each DD card, where nn (the third and fourth digits) is the data set reference number.

From the preceding definitions, it follows that it is the job of the person preparing the input data deck for a run of a cycle of M.E.T.R.O.-APEX to keep track of and specify the correct combination of team number on the Main Header Card and data set names (DSN=) for that team for data sets 15 and 16 on the corresponding DD cards. For this purpose it is recommended that the naming convention used in creating data sets for new teams follow some easily remembered and coded pattern, such as FILE15.TEAM2 as used in the preceding examples.

As previously noted, any number of teams can be run concurrently. However, the team numbers must be from 1 to 5 since five teams are the maximum that can be accomodated by data set 18.

It is extremely unlikely that any organization using M.E.T.R.O.-APEX will have any need for more than five teams, but in the rare event that it should become necessary the following notes are included:

For each additional set of five teams one additional data set "18" will be required. You will then need to become more sophisticated in data set naming. It is suggested that you use the convention of adding a letter to the data set names for the additional 15, 16 and 18 required:

FILE18A and FILE18B

FILE15.TEAM1A and FILE15.TEAM1B...FILE15.TEAM5B, etc.

Remember that the team number on the Main Header Card must still be only a digit from 1 to 5; no alphabetics are allowed. It becomes extremely important to keep good records and to carefully check the DD cards in the Job Control set for proper correspondence between data set names and team number. There is no protection in the main program to prevent using the files for TEAM1A when TEAM1B was intended, and vice versa.

Chapter 6

M.E.T.R.O.-APEX Data File Organization and Description

Introduction:

The disk "files" for M.E.T.R.O.-APEX contain and are used for three basic types of "data":

1. Actual land use, budget, population, etc., tables used to initialize the game and which are then updated from cycle to cycle as play progresses.
2. Tables of constants, conversion factors, and alphabetic character strings used for output.
3. Arrays of values saved temporarily during a cycle to make room for other data in core and then retrieved for further processing.

File Organization:

The files are organized by number into six direct data sets. The term "file" as used in this volume refers to a consecutive group of records within a data set which have the same length and variable attributes or are otherwise logically related. The six data sets have specific usages:

1. Data set 14 consists of fifteen "files" containing the "cycle 0" data base; that is, the tables and other variables representing the initial state of the game. It is referred to only when starting a NEWGAME by running cycle 1. The files in data set 14 are created by and, if necessary, are modified only by the FILEPRNT utility program. None of the files in data set 14 are ever written into by the main METAPEX program.

2. Data set 15 is the "working" data set during the processing of a cycle of the game. That is, it contains the accounts and stores of the various game roles which are added to or subtracted from or otherwise modified by the main section of the METAPEX program. It consists of 18 "files", the first 15 of which correspond to the 15 files in data set 14, the last three files are "temporarily-permanent"; that is, they are used only during the processing of a single cycle but are retained for possible value in tracing errors if any should occur.
3. Data set 16 consists of 15 files and is exactly analogous to data set 14 except that it contains "preceding cycle" records. Data set 16 is not changed during the run of a cycle, but is retained for comparison with data set 15 at the end of the cycle.
4. Data set 17 consists of five files containing constant data which is "read-only" by the main program. Included in this set of files is the list of capital projects and special programs which may be modified by the FILEPRNT program.
5. Data set 18 consists of sixteen files which, with one exception, are used only by the issue, candidate election and newspaper processing section of the main program. The one exception is the list of exogenous industries which is used by both sections of the program. Several of the files within this dataset are subdivided into groups of records sufficient for recording up to five game runs (teams) concurrently.
6. Data set 19 consists of three files containing issues, alternatives and newspaper headlines. It is a read-only data set.

Chapter 4, Error Messages, discussed the interaction of data sets 14, 15 and 16 during a game run. Because of its importance it will be repeated here. At the beginning of a game run, when either NEWGAME, CYCLE=1 or REPEAT=1

is specified on the Main Header Card, the fifteen files in data set 15 are copied into both data sets 15 and 16. When this copy operation has been successfully completed the program prints the message:

DATASETS 15 AND 16 HAVE BEEN INITIALIZED FOR NEW GAME

At this time the cycle number in bytes 1 and 2 of record 71 of each of these three data sets is zero.

At the end of a normal termination of the main part of the program (indicated by either going on to the newspaper and issue section or by a normal termination message if the newspaper is suppressed) the cycle number in data set 15 is set equal to the number of the cycle just completed.

When CYCLE= i (i greater than 1) is specified on the Main Header Card data set 15, the record of the last cycle completed, is copied into data set 16 and when the copy operation has been successfully completed the following message is printed:

STATUS AT END OF CYCLE i HAS BEEN COPIED FROM DATASET 15 TO DATASET 16 FOR START OF NEW CYCLE.

When REPEAT=i (i greater than 1) is specified on the Main Header Card data set 16, the record of the cycle preceding the last one completed, is copied into data set 15 and when the copy operation has been successfully completed the following message is printed:

DATASET 16 HAS BEEN COPIED TO DATASET 15 FOR REPEAT OF CYCLE i

Notice that at the beginning of the main section of the program data set 15 and data set 16 both contain exactly the same information.

File Record Length and Format:

The following six DEFINE FILE statements are included in both the root section of the main METAPEX program and the FILEPRNT utility program.

```
DEFINE FILE 14 (225,160,U,KX14)
DEFINE FILE 15 (305,160,U,KX15)
```

```
DEFINE FILE 16 (225,160,U,KX16)
DEFINE FILE 17 (358,94,U,KX17)
DEFINE FILE 18 (986,116,U,KX18)
DEFINE FILE 19 (2100,30,U,KX19)
```

Not all of the records within each data set utilize the maximum record length specified in the above define file statements. Where there is a discrepancy it will be noted in the discussion of each individual "file."

As noted earlier, each "file" is a group of consecutive records within a data set. In order for the program to reference the records within a "file" from 1 to n each direct access READ or WRITE statement within the program uses the algorithm (n+NRxx) where NRxx is an offset value for that "file" which is one less than the first record of "file" xx within the data set. These offset or base record numbers are set in a BLOCK DATA subroutine and communicated to the program in a named COMMON block, /FILES/ .

Each "file" is individually numbered for reference by the program and within this description. Table 1 is a cross-reference index to file numbers, data set reference numbers and records within a data set.

Table 1

CROSS-REFERENCE LIST OF M.E.T.R.O.-APEX FILES

File Number	Is in Data Set	Records	Maximum Record Length (Bytes)
31	14	1-58	640
34	14	59-64	640
37	14	65-67	640
38	14	68-69	640
35	14	70	328
36	14	71-78	80
39	14	79-81	640
30	14	82-98	640
32	14	99-100	640
61	14	101-163	96
62	14	164	624
64	14	165-171	464
68	14	172-223	56
531	14	224	348
532	14	225	134

Table 1 (continued)

File Number	Is in Data Set	Records	Maximum Record Length (Bytes)
1	15	1-58	640
4	15	59-64	640
7	15	65-67	640
8	15	68-69	640
15	15	70	328
16	15	71-78	80
19	15	79-81	640
20	15	82-98	640
22	15	99-100	640
51	15	101-163	96
52	15	164	624
54	15	165-171	464
58	15	172-223	56
511	15	224	348
512	15	225	134
513	15	226-254	636
12	15	255	640
11	15	226-305	640
41	16	1-58	640
44	16	59-64	640
47	16	65-67	640
48	16	68-69	640
45	16	70	328
46	16	71-78	80
49	16	79-81	640
40	16	82-98	640
42	16	99-100	640
71	16	101-163	96
72	16	164	624
74	16	165-171	464
78	16	172-223	56
541	16	224	348
542	16	225	134
3	17	1	232
21	17	2	376
6	17	3-162	72
28	17	163-278	58
53	17	279-358	80
204	18	1-10	64
205	18	11-20	12
206	18	21-30	40
207	18	31-35	290
208	18	36-40	120
209	18	41-88	56
210	18	89-688	240
211	18	689-698	256
212	18	699-703	464

Table 1 (continued)

File Number	Is in Data Set	Records	Maximum Record Length (Bytes)
217	18	704-708	290
237	18	709-716	290
247	18	717-731	256
101	18	732-821	80
102	18	822-901	160
103	18	902-981	28
104	18	982-986	32
201	19	1-300	88
202	19	301-600	76
203	19	601-2100	120

In the description of each file which follows, the "words" referred to are two bytes long, the length of all integers in M.E.T.R.O.-APEX, or an IBM 360 half-word. The type of variable is abbreviated I for Integer *2 and R for Real*4 unless otherwise specified. The element numbers given are for use in making file changes. For integer variables the element numbers and word numbers are the same. For real variables the element number is half the even word number.

For convenience, the file descriptions are in numeric order from 1 to 513. Since the first fifteen files in data sets 14, 15 and 16 have identical descriptions they are listed by the data set 15 number which is the first value in the following list:

Dataset 15	Dataset 14	Dataset 16
1	31	41
4	34	44
7	37	47
8	38	48
15	35	45
16	36	46
19	39	49
20	30	40
22	32	42
51	61	71
52	62	72
54	64	74
58	68	78
511	531	541
512	532	542

FILES 1, 31 and 41:

These files contain land-use ownership, value and price data which differs in each of the 29 M.E.T.R.O.-APEX analysis areas. A logical record for each analysis area requires two physical records, accounting for the 58 physical records in the file. The word count given is the sequence within each physical record. The following variables are for one logical record, they are identical for each analysis area.

Record #1

Words	Elements	Type	Name	
1-35	1-35	I	BDR (5, 7)	Number of developed units of each of the five residential housing types owned by each of the 7 game realtors. B (Game Realtors) Developed Residential Units By rules of FORTRAN storage the first 5 values (words 1-5) are for realtor 1; the next 5 (words 6-10) for realtor 2 and so on. The 5 values represent, in order of storage, the 5 developed residential zoning combinations in the following order: R1, R2, R3, M1, M2.
36-70	36-70	I	INDR (5, 7)	Same as above for up to 7 industrialists.
71-75	71-75	I	MDR (5)	Same as above for rest of private sector (market). Market Developed Residential Units.
76-80	76-80	I	PDR (5)	Same for the municipality (either real or simulated), depending on the analysis area.
81-85	81-85	I	CDR (5)	Same for county.
86-90	86-90	I	QR (5)	Empty residential units in each of the five categories. This is a subset of MDR; that is MDR includes both occupied and empty dwelling units. This array contains values as a result of emigration of families from an analysis area and is used to meet demands not met by realtors in the case of immigration or for sales to realtors and the public sector.

Words	Elements	Type	Name	
91-95	91-95	I	EMF (5)	Emigrated Families (formerly F array in METRO). Families leaving M (general market residential) matrix in the analysis area, by zoning type. Added to: (1) when market sells more developed residential units than are empty (QR array), i.e., occupied dev. residential; (2) when market developed residential land is rezoned, thereby causing it to become vacant. It is not initialized and is set to zero at beginning of each cycle.
96			IFIL	2-byte filler for boundary alignment.
97-180	49-90	R	BDC (6, 7)	Realtors developed commercial industrial and agricultural acres. B (Game Developers) Developed Commercial property in acres. Developed Non-residential Zoning: 1 - CL, Local Commerce 2 - CR, Regional Commerce 3 - IE, Endogeneous Industry 4 - IX, Exogeneous Industry 5 - OD, Exogeneous Bureaucratic 6 - AD, Developed Agricultural
181-264	91-132	R	INDC (6, 7)	Industrialists developed industrial and commercial.
265-276	133-138	R	MDC (6)	Market Developed Commercial and industrial property in acres. This array does include acres devoted to developed exogeneous commerce and industry.
277-288	139-144	R	PDC (6)	Public Developed Commercial.
289-300	145-150	R	CDC (6)	County Developed Commercial.
301-312	151-156	R	QC (6)	Empty developed "commercial." This is a subset of MDC (1), MDC (2) and MDC (3) in the same way QR is a subset of MDR.
313-318	151-159	R	P (3)	Public developed property: P (1) - Municipality P (2) - County P (3) - Right of way and streets
319-320	160	R	U	Base density factor for developed resi-

Words	Elements	Type	Name	
				dential. This value is the unit per acre for the least dense zoning category (R_1) in the particular analysis area. It is stored as the reciprocal. For example, 3 units/acre is stored as .3333 (1/3). (Multiplication factors for more dense zoning categories, which are constant across all analysis areas, are stored in array UDENV (5) in File 21.) The values are used to convert from units to acres and vice versa. ACRES = UNITS x UDENV (TYPE) x U UNITS = ACRES/ (U x UDENV (TYPE)) For example; 9 units of R_3 houses at a multiplier of 2 (.5) in an analysis area with a base factor of .33 would require 9 x .33 x .5 = 1.5 acres.

APEX File 1, Record #2

Words	Elements	Type	Name	
1-84	1-42	R	BVRC (6, 7)	B (Game Realtors') Vacant Residential and Commercial property holdings in acres. The six zoning categories, in order of storage are: 1 - Residential, single family R 2 - Residential, multiple family M 3 - Commercial C 4 - Industrial I 5 - Office O 6 - Agricultural A
85-168	43-84	R	IVRC (6, 7)	Same as BVRC for game - Industrialists.
169-180	85-90	R	MVRC (6)	Same as BVRC for non-game "market." This is the balance of all vacant land not in one of the other VRC arrays.
181-192	91-96	R	PVRC (6)	Same as BVRC for the municipality (public) in the AA.
193-204	97-102	R	CVRC (6)	Same as BVRC for county in the AA. Note: in some APEX routines, for example where we are summing all vacant

Words	Elements	Type	Name	
				land, these five arrays are combined and dimensioned as BVRC (6, 17). This is permissible since they are all stored consecutively by columns.
205-216	103-108	R	VVRC (6)	Value of Vacant Residential and Commercial Land in dollars per acre. These are the "appraised" values printed in the large table at the end and used for all sales of vacant property and assessment.
217-226	109-113	R	VDR (5)	Value of Developed Residential in dollars per unit of the 5 zoning categories, R_1 through M_2. This is the value used for assessing developed residential property and also the minimum sales price for sales of market developed residential units to game realtors.
227-238	114-119	R	VDC (6)	Value Developed Commercial (and industrial property in dollars per acre. Same as VDR except for the six commercial/industrial categories. See note above.
239-250	120-125	R	ROWF (6)	Right-of-way (streets, sidewalks, etc.) factor for each of the six categories of vacant land. The number of acres being developed is multiplied by the appropriate ROW factor to obtain the number of acres added to or subtracted from P (3) above. Note: Because right-of-way is always subtracted from vacant acres used for development, when developed land becomes vacant, the number of acres for right-of-way is taken from city and added to the vacant areas.
251-252	126	R	E	This is now a constant 1.0 for all AA's. EVEC (in file 21) contains cost per unit or per acre for each developed type. If at some time we conclude that improvement costs vary by AA, the factor could be changed.
253-254	127	R	ADDTB	Added tax base for exogeneous property.
255-276	128-138	R	PDRC (11)	Price of Developed Residential and Commercial property. These prices are only used for sales of game players' developed

Words	Elements	Type	Name	
				property. Dollars per unit for the first 5 values which correspond to the 5 price classes for single family and then multiple family residential. Dollars per acre for the last six values which correspond to the six classes of commercial, industrial and agricultural property the game players may develop. Initial 6 percent over corresponding value.
277-320				88 unused bytes available for expansion.

APEX File 3 - This is a single record "file" in data set 17 containing constants for use by the TOMM model.

Words	Elements	Type	Name	
1-58	1-29	R	ZH(29)	Maximum allowable density in persons per acre for each analysis area. This is used in the residential loop within TOMM model as a constraint on growth in an area
59-116	30-58	R	ATOT(29)	Total land area in acres for each analysis area. This is used in TOMM to compute the available land for commercial use by subtracting out all other uses.

APEX Files 4, 34 and 44 - These files each require six physical records. They are best described as government and schools financial data. The data from file 4 is retained in named COMMON block /FILE4/ during program execution.

Record #1

Words	Elements	Type	Name	
1-2	1-2	I	NKK(2)	Count of 10 word bond or millage records in sectors 4-5 (NKK(1)). NKK(2) is presently unused.
3-12	2-6	R	VTPP(5)	Total assessed value of all private property in each jurisdiction. Note that VTPP(5) is the sum of VTPP(1)...VTPP(4) and is total for county.
13-162	7-81	R	TVBK(5, 15)	Total appraised value of all private property for each of the seven land developers (1-7), the seven industrialists (8-14) and the general market (5) in each of the 5 jurisdictions.

Words	Elements	Type	Name	
163-192	82-96	R	BKVT(15)	Total value of vacant property for each of the 15 elements of the private sector.
193-222	97-111	R	BKDT(15)	Total value of developed property for each of the 15 elements of the private sector.
223-230	112-115	R	ASSF(4)	Assessment factor for each of the four jurisdictions.
231-248	116-124	R	TAM(9)	Total assessment millage for each of the following jurisdictions: 1-4 Municipalities 5 County 6-9 Simulated Schools This is used for computing tax due from private sector. It is the sum of AMS + SPECO + SPECD.
249-266	125-133	R	OPMIL(9)	Total operating millage. Sum of AMS + SPECO.
267-284	134-142	R	AMS(9)	Total Normal Operating Mills
285-302	143-151	R	SPECO(9)	Total of Special Operating Mills in effect
303-320	152-160	R	SPECD(9)	Total of all debt retirement millages.

Record #2

Words	Elements	Type	Name	
1-18	1-9	R	CLL(9)	Current Local Limit on operating mills.
19-36	10-18	R	SLIM(9)	State Limit on operating mills.
37-54	19-27	R	BDOTG(9)	Total General Obligation Bonds Outstanding.
55-72	28-36	R	GONEW(9)	Total New GO Bonds this cycle.
73-90	37-45	R	BDPGO(9)	Total Minimum Payments Due Next Cycle for GO Bonds.
91-108	46-54	R	DRDOL(9)	Total dollars in debt retirement fund at beginning of cycle. Generated from debt retirement millage + surplus from last cycle. Calculated for "this" cycle in last cycle's output links.
109-126	55-63	R	DRFGF(9)	Dollars for debt retirement transferred from general fund in current cycle.
127-144	64-72	R	CPFND(9)	Total dollars in Capital Fund. At begin-

Words	Elements	Type	Name	
				ning of cycle the value in file 44 is that remaining from preceding cycle. This is a continuous total reflecting additions from new bonds and transfers from general fund less amount disbursed to pay for projects.
145-162	73-81	R	TNEWB(9)	Total new bonds this cycle. Sum of approved GO bonds and all revenue bonds.
163-180	82-90	R	CFTRN(9)	Funds transferred to capital budget from general fund this cycle.
181-190	91-95	R	BDOTR(5)	Revenue Bonds Outstanding (no schools).
191-200	96-100	R	BDPRV(5)	Total Payment due on revenue bonds next cycle.
201-210	101-105	R	CREDP(5)	Percentage of state equalized valuation which determines GO bond limit.
211-220	106-110	R	BLIM(5)	GO bond limit.
221-229	221-229	I	CREDS(9)	Credit Rating for each jurisdiction. Integer values 1 to 3.
230		I	IWAST	Filler for boundary alignment.
231-236	116-118	R	RATEI(3)	Interest rate on new GO bonds for the three credit ratings.
237-238	119	R	REDUX	Reduction factor applied to reduce CREDP when credit rating drops.
239-320				Unused.

File 4, Record #3

Words	Elements	Type	Name	
1-160	1-80	R	OPVEC(16,5)	Operating budget percentages or dollars of the 4 municipalities and county (#5) carried from preceding cycle in file 44 and to next cycle in file 4. Value will be negative if a percent and positive if dollars. Columns are as follows for Municipalities (1-4) and County (5): 1 - Legislative, executive (Note: this item will be carried as a dollar value which cannot drop. It may only be increased by player input. 2 - Planning

Words	Elements	Type	Name	
				3 - Financial 4 - Judicial 5 - Fire and Police (Public safety for county) 6 - Parks and Recreation 7 - Water and Sewage 8 - Refuse collection 9 - Streets 10 - Employees benefits 11 - Public relations 12 - Library 13 - Public health (county only) 14 - APCO (county only) 15 - Welfare and hospitalization (county only) 16 - Unused
161-320	81-160	R	STPCV(16,5)	Standard per/capita dollar values for each budget expenditure item. Analogous to OPVEC. Zero if does not apply.

File 4, Record #4

Words	Elements	Type	Name	
1-18	1-9	R	TCYRV(9)	Total General Fund Revenue.
19-36	10-18	R	SURPL(9)	Surplus (or deficit) for each jurisdiction.
37-54	19-27	R	GENFT(9)	General Fund Property Tax revenue.
55-64	28-32	R	OTHAG(5)	Revenue from other agencies, in dollars. For county is services revenue.
65-74	33-37	R	POTH(5)	Percent change in other agency revenue each cycle.
75-84	38-42	R	FFLRV(5)	Dollar revenue from licenses, fees and fines.
85-94	43-47	R	PLFF(5)	Percent change in license, fee and fine revenue.
95-104	48-52	R	ONTR(5)	Dollar revenue from other non-tax sources.
105-114	53-57	R	PONT(5)	Percent charge in other non-tax revenue.
115-134	58-67	R	SPGRT(5,2)	Special grant revenue. i, 2 = Capital project fund grants i, 1 = General fund grants
135-136	68	R	RCREV	Road commission revenue for county.
137-138	69	R	PRCV	Percent charge in road commission revenue.

Words	Elements	Type	Name	
139-156	70-78	R	TGOPY(9)	Total payments made on General Obligation bonds current cycle.
157-174	79-87	R	SURPL(9)	Surplus in debt retirement fund after all bond payments.
175-176	88	R	DRINT	Interest paid on debt retirement surplus.
177-320				292 bytes unused.

Records 5 and 6 (640 words) contain a record of Government (politicians') bonds and special millages which are in effect. Each bond or millage requires a 10 word block, the contents of which are described below. Since the 10 word blocks contain four integer variables and three real variables, the 640 word array is dimensioned twice and equivalenced as follows:

```
DIMENSION KSMR (10, 64), SMR (5, 64)
EQUIVALENCE (KSMR (1), SMR (1))
```

Within this array new bonds or millages are added to the bottom of the list; and each cycle, as bonds or millages expire, the table is packed so that the oldest (first added) are always at the top of the list.

The 10 elements of each description block are as follows:

KSMR(1,I) - Jurisdiction; municipalities are 1-4. County is 5. Schools 6-9.

KSMR(2,I) - Bond or millage number.
Must be a three-digit integer starting with the cycle number times 100. For example, 301, 302, etc. for cycle 3.

KSMR(3,I) - Type (ISPEC) code:
1. General Obligation bond
2. Revenue bond
3. Special Operating millage
4. Debt Retirement millage

KSMR(4,I) - Years to run.

SMR(3,I) - Dollar amount of bond first year, unpaid balance after that.

SMR(4,I) - Interest rate (as, for example, 6.25) for bonds or number of mills.

SMR(5,1) - Minimum payment due on bond next cycle (calculated in output link and carried to next cycle) or temporary storage of bond over or under payment (calculated in BDMIL and saved here until output link).

M.E.T.R.O.-APEX File #6

This file contains the details of up to 120 capital plant projects and 40 special programs defined for M.E.T.R.O.-APEX politicians. The FILEPRNT utility program may be used to add projects or programs to the file, using a MAKE 6 control card. The first 120 records are capital plant projects and records 121-160 are special programs.

Each 72 byte record (36 "words")--one project--is made up in the following manner for capital projects:

Word	Type	Name	
1	I	LOC	Physical location, scale and restriction of project. Most projects will not be restricted, but a few which are expansion of existing facilities, may be restricted to a specific analysis area (there will be no ward or jurisdiction restrictions). Values 1-29 indicate location restricted to the corresponding AA. 30 - May be located in any analysis area. 31 - Must be located in an entire ward. 32 - Must be located in an entire jurisdiction.
2	I	ITYPE	Budget category 1 - Streets 2 - Sewers 3 - Water 4 - Parks and recreation 5 - Miscellaneous
3	I	IMPACT	CPI Impact of the project 0 - Analysis area 1 - Ward 2 - Jurisdiction 3 - County

Note: Player input will be by jurisdiction (Pols 1-4, County = 5). If location of project (item #1) not in player's jurisdiction, then the cost--impact on CPI--will apply to that player's jurisdiction,

Word	Type	Name	
	not to the jurisdiction where located.		
4	I	NYRS	Years required to construct project.
5	I	NBOND	0 - May not be financed by revenue bond. 1 - May be financed by revenue bond.
6	I	NISS	Issue number which will be printed in newspaper when planner recommends the project. (currently unused)
7-8	R	ACRES	Acres required, if any.
9-10	R	CMIN	Minimum cost (total).
11-12	R	CMAX	Maximum cost (total).
13-36	I	KTITL (24)	45 character project title (last three bytes of array not used).

Each 36 word record for Special Programs is formatted as follows:

Word	Type	Name	
1	I	LOC	Same as preceding records, except 33 = County-wide program.
2			Not used.
3			Not used.
4	I	NYRS	Number of years program is to run.
5	I	NCPRJ	Capital plant project also required, if any. To be started at minimum cost. Cost of each program is jurisdiction-wide.
6	I	NISS	Related issue to be printed when program has one year to go.
7-8			Not used.
9-10	R	CMIN	Cost per year.
11-12			Not used.
13-36	I	KTITLE (24)	48 character description of program.

M.E.T.R.O.-APEX Files 7, 37 and 47

These files contain records of capital plant projects in effect, recommended by planners or not carried by politicians.

The first record of each file is a record of politicians (government) projects and the second contains those projects and programs recommended by the planners during the current cycle for inclusion by the politicians' next cycle. The third record is discussed on a following page.

The array name usually used for the first sector is KAPLT (5, 63) and for the second KPLAN (5, 63). The first five words of each of the first two records are reserved for counters, etc., of the projects actually stored. Each project record consists of a five word sequence; that is, words 6, 7, 8, 9 and 10 of each sector are a record of a project; words 11, 12, 13, 14 and 15 are the record of another project, and so on. Only words 6-320 of each record are used for project records.

The five words in each project record are defined as follows:

Word	Example	
1	KAPLT (1, I) or KPLAN (1, I)	This word is set non-negative when a project is added to a list. It is equal to the associated bond number if it is tied to a bond or to zero if not (all planners' recommendations, of course, carry a zero code). The years the project is to run (4th word) is decremented by one in the output subroutine and, if the years remaining is zero, this first word is set equal to -1. (For planners' recommendations, record 2, this word is set to -1 in CPI if a recommendation matches a politician's input.) Politicians project lists are "packed" at the beginning of CPI each cycle to move all multiyear projects, i.e., projects for which the first word is non-negative, to the head of the list before any new projects are added.
2	KAPLT (2, I) or KPLAN (2, I)	Project number * 10 + CPI Impact
3	KAPLT (3, I) or KPLAN (3, I)	Location of project * 100 + Jurisdiction inputting (to force county projects to print in county budget). Location code is as follows: 1-29 - specific analysis area 30 - Ward 1 31 - Ward 2 32 - Ward 3 33 - Ward 4 34 - Ward 5 35 - Ward 6

Word	Example	
		36 - Jurisdiction 1 37 - Jurisdiction 2 38 - Jurisdiction 3 39 - Jurisdiction 4 40 - County
4	KAPLT (4, I) or KPLAN (4, I)	Years project is to run, if new. Years remaining after the first year of a multiyear project.
5	KAPLT (5, I) or KPLAN (5, I)	Cost per year in thousands of dollars. In other words, a project cost of $325,000 would be stored as the integer 325. If, due to division, a cost is not an even 1000, it is rounded up to even 1000s.

The first five words of each record are used as follows:

1	NCP	Count of the number of projects in the list in the corresponding record at any time.
2		Unused.
3-4	PJCST	Total dollar value of the projects in effect in the list.
5		Unused.

The third record of each of these files is used for two purposes:

(1) A record of special programs in effect.
(2) A record of those projects and programs recommended by the planners the preceding cycle but not put into effect by the politicians.

Unlike the first two records, the third record is divided into four word sub-records (since cost is not needed).

The first four words of the third record are:

1	I	NSP	Number of special programs in the first list.
2	I	NOP	Number of project records in the second list.
3-4	R	PGCST	Total dollar cost of special programs.

Words 5-164 are array KPROG (4, 40). The use of the first four words is the same for each special program as the first four words of each project descriptor in sector 1. The first word may be the number of a special operating millage.

Words 165-320 are array KNOPR (4, 39). Only the first three words are

pertinent and are as described for records 1 and 2.

M.E.T.R.O.-APEX File 8, 38, 48

This is a file of two 640 byte records used by the planner's output link during a cycle. The social indicators--unemployment, low income families, non-white population, deteriorating buildings, etc.--are calculated and ranked for all 29 analysis areas. Note that the total 908 byte logical record spans two physical records.

Words	Type	Name	
1-29	I	KPOPL(29)	Total population, number of persons.
30-58	I	NUNEM(29)	Number of unemployed.
59-87	I	RNKEM(29)	Unemployment rank--largest percent unemployment ranked as #1.
88-116	I	RNKLO(29)	Rank from 1 (highest) to 29 (lowest) percent in low income families.
117-145	I	RDETR(29)	Rank from 1 (highest) to 29 (lowest) percent deterioration.
146-174	I	RNKNW(29)	Rank from 1 (highest) to 29 (lowest) percent non-white.
175-232	R	PUNEM(29)	Percent unemployed.
233-290	R	PLOIN(29)	Percent of families with income below $3000/year. (Household type 5)
291-348	R	DETER(29)	Percent deteriorating residential structures.
349-406	R	PNONW(29)	Percent non-white population.
407-454	R	HP(6, 4)	Percent of people in each of the six wards perferring each of 4 fuel types--used by AIR model.

M.E.T.R.O.-APEX File 11

This file is used to store "STUFFed" records of input decisions and special output calculations.

Each "STUFF" is a record 16 words long which may contain a combination of real and integer elements depending on the type of STUFF. Therefore, each 320

word physical record can store up to 20 STUFF records. Since the file contains 50 physical records a total of 1000 STUFFS can be made each cycle.

The entire STUFF and search mechanism, including type codes, will be covered in a separate report.

M.E.T.R.O.-APEX File 12

This is a single 320 word file used for temporary or working storage between subroutines at several different points in the program.

It should not be used for other purposes between the points listed unless care is taken to read and restore the number of words indicated at the beginning of the record.

Many of these variables are retained in named COMMON block /FILE12/ during the main section of the program.

A. In the READA subroutine the following arrays are accumulated as BS, RM and AC cards are input--a total of 72 words--and saved until after the calculation of voter turnout in TRNOT.

Words	Name	
1-10	TTNSM(5)	Total non-school millage
11-18	TTSCM(4)	Total school millage
19-28	TTNED(5)	Total non-school dollars in bond proposals
29-36	TTSCD(4)	Total school bond proposals--dollars
37-54	RMRCY(9)	Requested millage rate, current year
55-62	RAF(4)	New assessment factors, if any
63-72	SCAMP(5)	Sum of campaign contributions, by jurisdiction.

B. Saved during TOMM model.

73-246 ERRM(29,3) - Endogenous employment prior to growth.

Saved during TOMM for comparisons at end.

C. Saved from the end of BDMIL to CPI:

73-111 NWORK(39) - Numbers of bonds and/or millages passed in BDMIL.

112 NPASS - The number of useful elements stored in NWORK

D. Newspaper Flags

Word 121 - Bits set to indicate low per capita budget items for central city.

Word 122 - Bits set to indicate low per capita budget items for county.

M.E.T.R.O.-APEX File 15

Arrays carried from main program to newspaper section.

Words	Type	Name	
1-16	R	CMPDL (8)	Total campaign dollars for each of 8 politicians.
17-32	R	OPDOL (8)	Total campaign dollars opposed to 8 politicians.
33-72	R	DOL (20)	Campaign expenditures (algebraic sum) for each of 20 bond or millage proposals.
73	I	NBS	Number of bond or millage proposals this cycle.
74-93	I	LBS (20)	Numbers of proposals.
94		filler	
95-104	I	PROMO (5)	Total promotional dollars for each of 5 jurisdictions.
105-114	R	PCSTD (5)	For CEM - Numbers of voters by ward.
145-164	I	ISFLG (20)	Numbers of issues to be printed in newspaper when special program has only one year to go. Stored as (issue # + 1000 x jurisdiction).

M.E.T.R.O.-APEX Files 16, 36 and 46

These are key files in M.E.T.R.O.-APEX.

Only the first 3 records of the eight possible records are currently being used. The first represents the first 40 words of blank COMMON storage. Since these words contain some pointers and indicators for STUFF and for searching the STUFFed file (11) as well as other common information, they must be carefully saved and restored if this upper common area is ever to be used for other purposes, and must also be saved at various points for restarts.

The use of the forty words is as follows:

Word	Type	Name	
1	I	NCYCL	Cycle number. Set to 0 in file 36. Incremented by 1 in core if going on to new cycle, but cycle number not incremented in the file until end of suceessful execution.
2-4	I	KDATE(3)	Date of run. Set in MAIN for use by EJECT in output links. First word is month, second is day, third is year.
5	I	LDP	The first seven bits (0-6) are set by ISET to indicate those realtors playing in the CYCLE. Bits 7-13 are set for playing industrialists.
6	I	LRAND	Initial 5 digit random number initial value. Set to 24683 for cycle 1. Thereafter it is the "old" value plus the sum of the 3 elements of KDATE and adjusted to be odd number.
7-10	I	MIN(4)	These sixteen words are used for STUFFing and
11-22	R	ZIN(6)	searching. The first four words are always integer * 2, sometimes referred to as the first four elements of MIN(16) but also referred to by specific names in some links. The next twelve words may be either integer * 2 or real * 4, depending on the STUFF record. They are usually referred to as ZIN(6). MIN and ZIN are equivalenced as follows: EQUIVALENCE (MIN(5), ZIN(1)) Other specific variable names are frequently equivalenced to specific elements of MIN or ZIN. See STUFF description for more details.
23	I	PACKI	The number indicating which of the 20 sixteen word STUFF records in a physical record was the least one STUFFed.
24	I	PACK2	Which sector of the 50 in file 11 was last STUFFed.
25	I	IXPRK	Index of "found" STUFF from subroutine QERCH.
26-29	I	INFO(4)	See comments in QERCH.
30	I	LPK1	QERCH analogues of PACK1 and PACK2.
31	I	LPK2	
32	I	KC44	Player number (developer, industrialist, politician, etc.). Used by EJECT for page headings.
33	I	IROLE	Flag for EJECT to indicate output heading. 1 - Realtor 2 - Politician

Word	Type	Name	
			3 - Educators 4 - General Summary 5 - APCO 6 - Industrialist 7 - Planners
34	I	IFUSE	Flag set in output subroutines, primarily miscellaneous output, to decide if number of lines printed requires skipping to the top of a new page. Used only in output links.
35	I	NPAGE	Used by EJECT subroutine in output links to keep running page number.
36	I	LSW	Switch set for print suppression of individual players in output links by data switch setting. = 1 Print = 2 Don't print
37	I	NOIO	Master print suppression flag. Set in response to SUPPRESS=ALL to suppress all printing.
38	I	NTEAM	Team number.
39-40			Temporary use in LAND section of program.

Record #2

Word	Type	Name	
1-28	R	CASH(14)	Running cash balance in account of each of the realtors (1-7) and industrialists (8-14).
29-40			Not used at present.

Record #3

Word	Type	Name	
1-20	R	TNAME(10)	Team name, EBCDIC 10A4.
21-40			Unused.

M.E.T.R.O.-APEX Files 19, 39 and 49

These files contain capital plant values, indices, ranks, population equivalents. One logical record of 1888 bytes spans three physical records of 640 bytes.

The following is a summary of the file 19 contents; a detailed variable de-

scription follows the summary:

Consecutive Words	Variable Type	Name and Dimensions	Physical Record Number	Physical Record Word	Elements
1-648	R	CPTOT (36, 9)	-	-	-
		The 9 columns of the above array are as follows:			
1-72	R	Col. (1) Streets: CPTOT (1,1) to CPTOT (36, 1)	1	1-72	1-36
73-144	R	Col. (2) Sewers	1	73-144	37-72
145-216	R	Col. (3) Water	1	145-216	73-108
217-288	R	Col. (4) Parks/ Recreation	1	217-288	109-144
289-360	R	Col. (5) Miscellaneous	1 2	289-320 1-40	145-160 1-20
361-432	R	Col. (6) Total Government (sum of first 5 columns.)	2	41-112	21-56
433-504	R	Col. (7) Total Elementary School	2	113-184	57-92
505-576	R	Col. (8) Total High School	2	185-256	93-128
577-648	R	Col. (9) Total School (sum of 7 & 8)	2 3	257-320 1-8	129-160 1-4
649-792	R	CPI(36, 2)	3	9-152	5-76
793-864	R	POPEQ(36)	3	153-224	77-112
865-866	R	CONVF	3	225-226	113
867-938	I	KPRNK(36, 2)	3	227-298	227-298
939-940	R	COCPT	3		
941-942	R	COCPI	3		
943-944	R	COPOP	3		
945-960		Unused at present.			

Variable Descriptions for Files 19, 39 and 49:

CPTOT (36, 9) — Total capital plant value in each of 9 categories (see below) for each of 36 geographic areas. In actual practice the program adds cost per year from file 6 as each project is processed in SUBROUTINE CPADD. Once a project is completed no appreciation or depreciation on value takes place.

The first 29 rows correspond to the 29 APEX analysis areas.

Row 30	Ward 1	Total
Row 31	Ward 2	Total
Row 32	Ward 3	Total
Row 33	Ward 4 or Jurisdiction 2	Total
Row 34	Ward 5 or Jurisdiction 3	Total
Row 35	Ward 6 or Jurisdiction 4	Total
Row 36	Jurisdiction 1, Central City	Total

CONV — Conversion factor for converting number of employees to population equivalents, POPEQ. It is presently (3-31-7) equal to 1.25. Population equivalents are calculated as:

POPEQ (i) = Total Employees in all categories / CONVF

CPI (36, 2) — Capital Plant indices for each of the 36 areas described above. Column one is government (politicians) and is column 6 of CPTOT/POPEQ. Column two is elementary school capital plant index:

$$CPI(I,2) = \frac{CPTOT(I, 7)}{\text{Number of elementary pupils}}$$

KPRNK (36, 2) — The rank of the corresponding capital plant indices--the largest value is ranked 1.

COCPT — Total dollar value of county capital plant (miscellaneous).

COCPI — County capital plant index.

COPOP — County population equivalents.

M.E.T.R.O.-APEX Files 20 (30 and 40)

This seventeen record file contains records of the seven game realtors' and seven industrialists loans and taxes as well as other arrays and constants necessary in subroutine TAXLN for loan and tax processing.

Each of the first 14 records in these files are only 80 bytes long and contain variables pertaining to one of the game realtors (records 1-7) or industrialists (records 8-14). The variables in each of the records are as follows:

Word	Type	Name	Description
1-2	R	OLDCH	Cash balance at end of preceding cycle for the player.
3-4	R	PROPV	Total property value for the player at end of preceding cycle.
5-6	R	TTXPD	Total taxes paid current cycle for the player.
7-16	R	TAX(5)	Taxes for the player for each jurisdiction computed at end of preceding cycle or total unpaid delinquent taxes during current cycle after payments are processed.
17-26	R	TDELT(5)	Total delinquent taxes paid during current cycle in each jurisdiction by the player due to land confiscation.
27-31	I	DELTX(5)	Counters of number of times taxes were not paid in full for the player, in each of the 5 jurisdictions. If unpaid for two consecutive cycles, land is confiscated for the payment.
32	I	LSF	Financial standing of the player from 1 (highest) to 3 (lowest). Presently based only on loan underpayments.
33-34	R	TLDBT	Total outstanding loan balance for player.
35-36	R	TNEWL	Total dollar value of new loans current cycle for the player.
37-38	R	TLPAY	Total loan payments made current cycle by the player.
39-40	R	FLLMT	Loan limit factor for the player--the proportion of his net worth which he can borrow.

The 15th record of these files is 128 bytes long and contains variables not specific to any one player but which are used for calculations relating to all players:

Word	Type	Name	Description
1	I	NJ	Count of number of loans in the loan table--which

Word	Type	Name	Description
			follows--or pointer to last used 10 word block.
2	I	NEXT	The last loan number assigned. This is, in effect, a count of the total number of loans processed during a game since loan numbers are not re-used once a loan is paid off. It does not necessarily represent the number of loans in the table.
3-14	I	IRATE(4,3)	Interest rates at which new loans are granted. The four rows are differences due to life (length) of the loan: IRATE (1,I) - 1 or 2 years IRATE (2,I) - 3 to 5 years IRATE (3,I) - 6 to 10 years IRATE (4,I) - over 10 years The three columns are for the 3 possible credit standings (LSF).
15-16	R	XINT	Interest rate paid on cash savings for players. Added at beginning of cycle to cash from preceding cycle.
17-18	R	TINT	Interest rate charged and added to unpaid taxes.
19-20	R	PENFC	Penalty factor against loan underpayments.
21-22	R	REDOX	Factor for reducing FLIMT if loan underpayment, i.e., reducing loan limit.
23-36	I	LUC(14)	Loan underpayment counters for each player. If delinquent in loan payments 3 consecutive cycles, payment is forced from cash balance.
37-64	R	TNETW(14)	Net worth at end of cycle for each player. Used primarily from file 40 for comparison with current value.

The 16th and 17th physical records of these files make up one logical record which is the table of loans outstanding for the realtors. Each loan requires a block of ten words (20 bytes) so that a maximum of 64 loans may be outstanding at any one time. (When the table becomes full during play the table is packed by eliminating all loans with a zero balance.)

The first four words of each 10 word loan record are 2 byte integers and the last three elements (6 words) are real * 4. For this reason the array is

identified by two names and equivalenced as follows:

```
DIMENSION XJ (5, 64)

COMMON J (10, 64)

EQUIVALENCE (J (1), XJ (1) )
```

The elements of each loan record are as follows:

Word		
1	J(1,I)	Number of the land developer or industrialist receiving the loan. Integer from 1-14.
2	J(2,I)	Loan number. Assigned by the program from 1-N, printed on the output, and used to identify loan payments.
3	J(3,I)	Interest rate. Note: this is an integer value, e.g. 81 = 8.1%.
4	J(4,I)	Years remaining in loan life.
5-6	XJ(3,I)	Dollar amount of loan or, after the first year, the unpaid balance.
7-8	XJ(4,I)	Dollar penalty for underpayment.
9-10	XJ(5,I)	Minimum payment next cycle.

M.E.T.R.O.-APEX File 21

A single record file containing constants used in land transactions in LAND and SELL. All values are preset and are not initialized or saved from preceding cycle.

Words	Type	Name	
1-10	R	UDENV(5)	Spread vector used with "U" variable (see File 1 description) to get increasing land use density.
11-22	R	DTS(6)	Proportion of total property value which is demolition cost when property goes from developed to vacant. Values for the six vacant categories.
23-24	R	PNF	Public N Factor - a proportion of public land which becomes right-of-way when public land is developed.
25-26	R	PDFS	Dollar per acre cost of public land demolition.

Words	Type	Name	
27-48	R	RCDEV(11)	Residential-Commercial development cost. Dollars per unit (or acre, if commercial) to construct or develop property for the 11 developed categories.
49-70	R	EVEC(11)	Raw land improvement cost (clearing, streets, curbing, sidewalks, etc.) for the same 11 categories as RCDEV. May be modified by E factor in File 1 which is presently at 1.0.
71-72	R	R2R2V	Proportion of multiple zoning type 2 (MDR(5) family or housing units which can be displaced by commercial demand in SELL. Presently set at 10%.
73-122	R	HPREF(5,5)	Percent of each household type (2nd subscript) that prefers each of the five developed residential housing types (1st subscript). Used to allocate demand after TOMM.

Housing Type

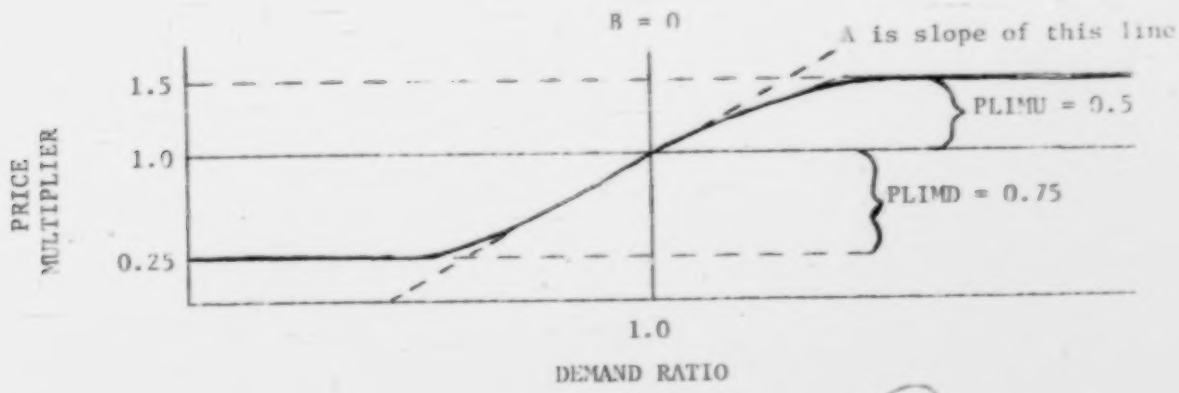
H.H. Type	R1	R2	R3	M1	M2	
1	50%	30%	0	20%	0	100%
2	20%	20%	20%	20%	20%	100%
3	10%	30%	20%	25%	15%	100%
4	0	20%	40%	10%	30%	100%
5	0	0	40%	0	60%	100%

Words	Type	Name	
123-172	R	ZPREF(5,5)	Percentage of each type of household living in each housing type. Used to allocate displaced households when units go from developed to vacant.

Household Type

	H_1	H_2	H_3	H_4	H_5	
R_1	62.5%	25.0%	12.5%	0	0	100%
R_2	30.0%	20.0%	30.0%	20.0%	0	100%
R_3	0	16.7%	16.7%	33.3%	33.3%	100%
M_1	26.7%	26.7%	40.0%	13.4%	0	100%
M_2	0	16.0%	12.0%	24.0%	48.0%	100%

Words	Type	Name	
173-174	R	PLIMU	
175-176	R	PLIMD	Related to price adjustment for residential property. See following discussion.
177-178	R	PA	
179-180	R	PB	
181-182	R	CLIMU	
183-184	R	CLIMD	Analogous to preceding four variables for commercial prices.
185-186	R	CA	
187-188	R	CB	
189-320			Unused at present.



Price adjustments each cycle are computed as a function of the ratio of demand for a particular type of property within an analysis area to the demand for the same type over the entire APEX area. The function generates a multiplier which is applied to the corresponding price and which is calculated from an S-curve algorithm. In other words, if demand is greater in a specific area than overall a multiplier such as 1.03 or 3% increase would be used. If demand were lower the multiplier might be a 0.96 or 4% decrease.

Mathematically the multiplier is calculated as follows:

If $T \geq 1.0$, then $X = T-1$ and $M = S * LIMU + 1.$

If $T < 1.0$, then $X = \frac{(T-1)}{T}$ and $M = S * LIMD + 1.$

Where T = demand ratio

M = price multiplier

$$S = \frac{2.}{e^{(-AX + B)} + 1.} - 1.$$

In simple terms, LIMU = maximum proportion of increase in price, i.e., how much greater than 1.0 can the multiplier be.
LIMD = maximum proportion of decrease in price, i.e., how much less than 1.0 can the multiplier be.
A = maximum slope of S curve, which determines how rapidly the multiplier increases or decreases with a change in demand ratio.
B = horizontal-axis intercept. It is set at 0.0 which causes a ratio of 1.0 to produce a multiplier of 1.0. If B > 0 the axis would be moved to the right, meaning that it would require a demand ratio greater than 1.0 to keep the price from going down as might be the case in a depression or, if B < 0 the axis would be moved to the left which would mean that a demand less than 1.0 could still cause an increase in price.

M.E.T.R.O.-APEX Files 22, 32, 42

Counters of families and employment.

Record #1:

Words	Type	Name	
1-145	I	PDM(29,5)	Number of families in each of 5 types in each AA.
146-232	I	NJGTC(29,3)	Number of employees in each of the thre endogenc employment categories in each AA. Col. 1 is local commercial; Col. 2 is regional commercial; Col. 3 is local industrial.
233-319	I	JDELT(29,3)	(Used only in file 22, not initialized.) Change in employment from TOMM--used in SELL.
320			Unused.

Record #2:

Words	Type	Name	
1-58	R	EBUR(29)	Number of exogenous bureaucratic employees in each analysis area.
59-116	R	EIND(29)	Exogenous industrial employees in each AA.
117-118	R	TEXMP	Total Exogenous Employment after GROW.
119-128	R	HFCT(5)	Persons per household by type.
129-140	I	EXBIT(12)	Same as words 54-65 of File 512. Set to zero in file 532.

M.E.T.R.O.-APEX file 28

This 116 record file (four seasons times 29 analysis areas) is a special matrix of wind diffusion data for the AIR model. The details of its calculation and use will be dealt with in a separate report.

M.E.T.R.O.-APEX Files 51, 61 and 71 Process File

Data on each industrial process for gamed and simulated industries.

Note: The record number of each process is indexed in the Industry File (#58). That is, process x for industry I is in record number IRECD + x - 1, e.g. process 2 for industry 4 would be in IRECD + 2 - 1 and if IRECD = 10, process 2 is in record 11 of the process file.

Word	Element	Type	Name	Description
1-10	1 5	R	REMIS(5)	Rate of contaminant emission current (in lb/hr).
11	11	I	RSMOK	Rate of smoke emitted (Ringlemann's).
12	12	I	RODOR	Rate of odor emitted (Stinks).
13-22	7-11	R	GEMIS(5)	Potential (uncontrolled) rates.
23	23	I	GSMOK	Potential Ringlemann.
24	24	I	GODOR	Ditto Stinklemann's.
25-27	25	I	ICS(3)	Control system applicable. Negative: not operating, not installed. .LT. 100: operating .GT. 100: installed, not operating.
28	28	I	ITYPE	Odor type: 1 = putrid, 2 - acrid, 3 = sulphurous, 4 - pungent.
29-30	15	R	RPROC	Process rate, in input mat./day, except for fuel, quid in lb/unit output.
31-35	31-35	I	MUNIT(5)	Name of units in RPROC. (10 characters)
36	36	I	IRATE	Conversion factor for RPROC. RPROC * IRATE = production rate in 1000 lb/hr.
37-38	19	R	OPER	Operating period, as % of day.
39-48	20-24	PNAME(5)		Process Name. (20 characters)

M.E.T.R.O.-APEX File 52, 62, 72 - Air Pollution Laws and APCO data

Word	Type	Name	Description
1-10	R	PRIOP(5)	Price of operation of 5 types of monitoring stations.
11-12			Unused.
13-84	R	PRICE(6,6)	Upgrade costs of monitoring stations: 0-5 x 0-5
85-86	R	COSTI	Cost of plant inspection.
87-88	R	COSTM	Cost of emission measurement.
89-100	R	CB(6)	County Budget Items.
101-102	R	CTOT	Total County Allocation.
103-108	R	FFUND(3)	Federal fund items for 3 years.
109-144			Unused.
145-173	I	OPMON(29)	Monitoring stations in operation.
174-202	I	MONS(29)	Monitoring stations in existence.
203-212	R	APASK(5)	Air pollution control officer's requests for contaminant level used to compute the number of days above some level.
217-246	R	PDSHR(5,3)	Pounds/hour limitation on the five pollutants--(1,i) = Particulates, (2,i) = SO_2, (3,i) = CO, (4,i) = NO_x, (5,i) = HC.
247-276	R	PDSPU(5,3)	Pound/processing unit limitation, as above.
277-279	I	IMOKE(3)	Smoke limit, Ringleman 0-5.
280-282	I	IODOR(3)	Odor limit, Stinkleman 0-5.
283-288	R	FINE(3)	Fines.

The three columns in arrays PDSHR, PDSPU, IMOKE, IODOR, and FINE are, respectively, state regulations, county regulations and recommendations to the state.

289-302	R	AMT(7)	Amounts gamed industrialists have been fined.
303-312	R	COEFF(5)	Coefficients for complaint generator.

M.E.T.R.O.-APEX File 53 - Control Systems

This file holds the percentages for reducing emission rates due to installation of the named system.

Word	Type	Name	Description
1-10	R	PCT(5)	Percent reduction in emissions - particulates, SO_2, CO, NO_x, HC.
11-12	I	LSMOD(2)	Reduction in smoke and odor.
13-14	R	ORIGD	Original cost of installation.
15-16	R	OPERD	Cost of annual operation.
17-40	I	NAME(24)	48 character control system name.

M.E.T.R.O.-APEX Files 54, 64 and 74 - Gamed industries data

The Industry number is the record number.

Words	Type	Name	
1-3	I	NEMP(3)	Number of employees last 3 cycles.
4	I	IAA	Analysis area of plant.
5-6	R	TEQBG	Total value Equipment, Building and Air Pollution Equipment.
7	I	IOP	1 if available for play. 0 if not available for play (no data).
8	I	MFUEL	Fuel used this cycle.
9-10	R	CINCP	Capacity increment due to invest in equipment (Units/$).
11-12	R	CPMIN	Minimum expenditure on equipment.
13-14	R	CINCB	Potential capacity increase due to building increase (units/sq. ft.).
15-16	R	BINVM	Minimum building investment increment allowable.
17-18	R	CLAND	Building capacity change due to land increase (sq. ft./ sq. ft.).
19-20	R	SINCB	Increment in sq. ft. building/$ expenditure on building.
21-22	R	EMPDR	Employment/Production ratio (employees/unit).
23-32	I	IPCTS(10)	% changes in sales standard MKTSL.
33-42	I	IPCTC(10)	% changes in PRMKT

Words	Type	Name	
43-52	I	IPCTC(10)	% changes in product-costs.
53-54	R	MCAP	Maximum capacity of firm next cycle.
55-60	R	NPL(3)	Production Level.
61-66	R	NESTS(3)	Estimated sales (unit).
67-72	R	NACTS(3)	Actual sales (unit).
73-78	R	PRICE(3)	Price set by industrialist. ($)
79-84	R	PRMKT(3)	Market price. ($)
85-90	R	ESTS(3)	Estimated sales. ($)
91-96	R	ACTS(3)	Actual sales. ($)
97-102	R	INSLD(3)	Inventory sold (units).
103-108	R	IVNT(3)	Inventory last 3 cycles (units).
109-110	R	ICOVR	Inventory carry over (units).
111-112	R	AVEWG	Average yearly wage/employee.
113-114	R	UMC	Unit material cost. ($/unit)
117-118	R	CCIVN	Total inventory carrying cost.
119-120	R	CPINV	Current investment in equipment (non-pollution). ($)
121-122	R	CPLIM	Equipment investment limit (maximum). ($)
123-124	R	BINVE	Current investment building. ($)
125-126	R	BDLIM	Building investment limit. ($)
127-128	R	DPREX	Depreciation expenditure (deterioration).
129-130	R	SQFTB	Sq. Ft. of building.
131-133	I	ICAP(3)	% capacity used.
134 I	I	ISHP	Flag = 1 to indicate purchase of control system data this cycle.
135-136	R	TLC	Total labor cost.
137-138	R	TMC	Total material cost.
139-140	R	TFC	Total fuel cost.

Words	Type	Name	
141-142	R	TAC	Total administrative cost.
143-144	R	TPCOP	Total pollution control operating expenses paid.
145-146	R	APFPD	Air pollution fines paid.
147-148	R	EMMEX	Emission measurement expenditure this cycle.
149-150	R	CAMEX	Campaign contributions.
151-152	R	DPAL	Depreciation allowance made.
153-154	R	PROG	Gross profits.
155-156	R	FEDS	Federal-state taxes.
157-158	R	PRON	Net profit.
159-160	R	CEXEQ	Capital expenditure on equipment (non-pollution).
161-162	R	CEXLD	Capital expenditure on land.
163-164	R	CEXBG	Capital expenditure on building.
165-166	R	CEXAP	Capital expenditure on air pollution.
167-178	R	CFUEL(6)	Unit cost of each of six fuels. ($/unit)
179-180	R	TLPAY	Total loan payments.
181-182	R	TTXPD	Total tax payments (local).
183-188	I	IFUEL(6)	Fuels available for use (1 = available, 0 = not available). The six fuels, are: Low grade coal, high grade coal, lowgrade oil, high grade oil, natural gas electricity.
189-190	R	UNIT	Name of product unit.
191-192	R	TAPCV	Total air pollution control value.
103-194	R	DVLPD	Acres developed land.
195-196	R	VACNT	Acres vacant land.
197-198	R	MKTSL	Market sales standard.
199-204	R	MAXC(3)	Capacity last 3 cycles.
205-206	R	UICC	Unit inventory carrying cost.
207	I	NPROC	Number of subprocesses.
208-222	I	NAME(15)	Industry name (30 characters).

Words	Type	Name	
223-223	R	TOTC	Total cost of production.
225-226	R	TCASH	Total cash available.
227-228	R	TCAP	Total capital expenditures.
229-230	R	AVAL	Total land value.
231-232	R	ALLOW	Maximum depreciation allowance for next cycle.
233-240			Unused.

M.E.T.R.O.-APEX File 58,68,78 - Record of all industries, gamed and simulated

Note: The record number corresponds to the Industry number.

Words	Type	Name	Description
1	I	ICP	Operational Status of Industry
2	I	IAA	Analysis Area.
3-4	R	RPROD	Output of Firm.
5-6	R	PRODC	Production Capacity of Firm.
7	R	IRECD	Record number in file 51 of the first process for gamed industries. Except for the gray iron foundry,(Ind.3) IRECD is the fuel process. For simulated industries, it is the only process.
8	I	NPROC	Number of production processes used by the Firm. For simulated industries, NPROC = 0.
9-13	I	NUNIT(5)	Name of appropriate production unit per unit of time.
14-28	I	FNAME(15)	Firm name--30 characters.

M.E.T.R.O.-APEX File 204

This file consists of ten 64-byte records, where each record corresponds to a cycle of the game. Each record is identical.

Words	Type	Name	Description
1-32	I	ISSUT(32)	Each non-zero value is the number of a "record" in file 201. These are the preset issues for the cycle represented by the record number. A listing of these preset issue numbers may be obtained by running FILEPRNT and specifying PRINT 204.

M.E.T.R.O.-APEX File 205

This file consists of ten 12-byte records. Each record corresponds to a cycle of the game from 1 to 10.

Words	Type	Name	Description
1-6	I	IEXO(6)	The numbers of from 1 to 6 exofirms which are preset to desire entry into the community the following cycle. Used only by the newspaper printing section of the program.

M.E.T.R.O.-APEX File 206

This file consists of ten 40-byte records, where each record corresponds to a cycle of the game. Each record is identical and consists of a set of from 1 to 20 preset state and national headlines. Since the one vector contains both national and state headlines and since they may be a combination of single-line heads and multipe-line heads the following scheme is used to identify the values in the file:

1. The first values in each record are national headlines. As the program interprets each record from left to right the first state headline is identified by being a negative value. All values following the negative number are state headlines. A zero value terminated the list.
2. Values less than 1000 are record numbers in the issue file (file 201). All values greater than 1000 are record numbers in the headline file (file 203) to which 1000 have been added.

M.E.T.R.O.-APEX File 209 (also referred to as file 5 in some instances)

This file contains descriptions of all of the preset exogenous industries which are programmed to wish to locate in the METAPEX area during the game. Only 45 of the 48 records are initially used allowing room for expansion of the file.

Each exofirm record consists of the following data:

Words	Type	Item	
1-16	I	NAME(16)	Name of the exofirm. Up to 32 characters, 2 characters per word (16As).
17-20	I	IPREF(4)	Analysis area in which the firm wishing to locate in order of preference. May be one to three given, fourth is unused.
21	I	NEMP	Number of employees.
22	I	IZONP	Coded to indicate the land use type which needed to be rezoned industrial. Now unused since the restriction has been abandoned.
23	I	NISSU	Number of issue generated if the firm enters the city.
24	I	NYRS	Years over which the realtors receive a return on their investment.
25-26	I	RATE	Rate of return on realtors investments, stored as 1.05 equiv. to 5%.
27-28	R	AMT	Amount of realtor investment required in order for an exofirm to enter.
29-30	R	ACRES	Numbers of acres of I_2 land needed for this exofirm.
31-32	R	TAX	Dollar addition to the tax base in the analysis area in which the firm locates.
33-34	R	COST	Cost of capital plant additions (sewer, streets) required for entry of this exofirm.
35	I	IBIT	Word containing bits set to indicate conditions. These bits are tested by function ITEST: Bit 0 - 1 if Industrial 0 if Office (bureaucratic) Bit 1 - On (=1) if firm should be forced in even though all politicians and/or realtors conditions are not met. Bit 2 - On if rezoning required. Bit 3 - On if new sewers required. Bit 4 - On if street improvements required. Bit 5-15 = Unused.
36	I	INDEF	If non-zero, number of industry in file 58 which must

M.E.T.R.O.-APEX Files 511, 531, and 541

This single record file contains vectors of preset growth percentages for various sectors of that local economy for ten cycles plus records of national and local employment and unemployment. (In the 1130 version of the program it was a single file included as part of file 3. It was made a separate file because it may vary from team to team.)

Words	Elements	Type	Name	Description
1-10	1-5	R	UNEMP(5)	Local unemployment for each of the five household types in current cycle. Calculated from UNAPX(t+1).
11-32	6-16	R	UNAPX(11)	APEX area unemployment rate. The subscript is cycle number +1. The first value is starting value for cycle 0, the second value calculated for cycle 1, etc.
33-34	17	R	STPPA	The starting, cycle 0, student population of the university presently set at 27,000.
35-54	18-27	R	STPOP(10)	A record of the new student population for each cycle. The value for the current cycle is printed in the output for Politician 2.
55-74	28-37	R	PCTNE(10)	Percent Change in Total National Employment.
75-94	38-47	R	PEXTC(10)	Percent Change in EXogenous employment in APEX This Cycle, both bureaucratic and industrial, for 10 cycles. This change including addition of exofirms and gamed industrial growth after national growth rate applied.
95-114	48-57	R	GOVTG(10)	The present growth proportion of exogenous bureaucratic employment (EBUR) in AA 8 (state government) for 10 cycles. A 3% growth would be stored in this array as 1.03.
115-134	58-67	R	UNIVG(10)	The growth proportion of university enrollment by cycle.
135-154	68-77	R	CUSLF(10)	Percent change in total US labor force.
155-174	78-87	R	CAPLF(10)	Percent change in labor force in APEX area.

M.E.T.R.O.-APEX Files 512, 532 and 542

These files contain records of which exofirms are preset to enter in up to 10 cycles, which exofirms did and did not enter and other information pertaining to exofirms for newspaper printing.

Words	Type	Name	Description
1-40	I	NEXOP (4,10)	This array is used by the GROW section of METAPEX to indicate those exofirms desiring to locate in the area in each of the 10 cycles. Each column of the array represents a single cycle. The four elements NEXOP(1,i) through NEXOP(4,i) constitute an array of 64 consecutive bits which are set on (1) by the ISET routine for those exofirms desiring to enter. Only six may be so set for any one cycle. All other bits are set off (0). They are tested in GROW by the ITEST routine (an alternate entry point in ISET).
41	I	NUMEX	The count in any single cycle of the number of exofirms which did enter.
42-47	I	NEXAA(6)	The number of the analysis area in which the firm in the corresponding element of NEXIN located.
48-53	I	NEXIN(6)	The numbers of the exofirms which entered in a cycle. For example, if NEXIN(1) = 12 and NEXAA(1) = 17 it means that exofirm 12 located in analysis area 17.
54-57	I	EXOUT(4)	A 64 bit vector, cleared at the beginning of GROW, in which the corresponding bit is set on by ISET for each exofirm which does not enter in a cycle.
58-61	I	EXTHS(4)	A similar vector of bits containing a bit set on for each exofirm which entered in the current cycle. It is the same information as that in NEXIN but in a different form.
62-65	I	EXIN(4)	A vector of 64 bits which is cleared only at cycle zero and has a bit set on for each exofirm entering during a game. In other words, a cumulative record of all exofirms which entered during a game.
66-67	I	IAIRQ(2)	Bit corresponding to Analysis area set on if flagged for air pollution headline in newspaper. Set in AIR.

Notice that each of the above arrays is explicitly defined as integer *2. An integer *2 array, EXBIT (12), is also defined in GROW and the above arrays are equivalenced as follows:

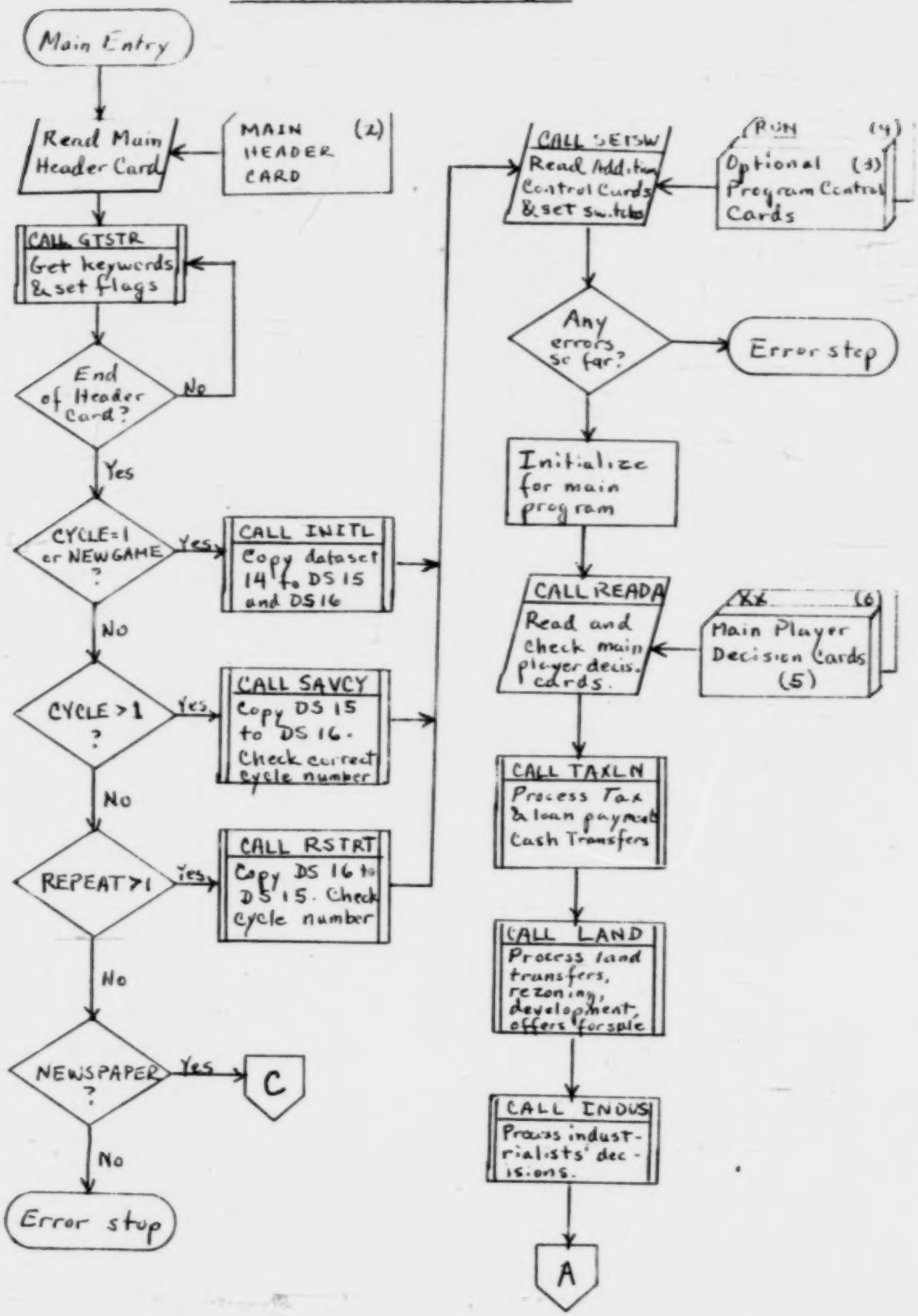
(EXBIT (1), EXOUT (1)), (EXBIT (5), EXTHS(1)), (EXBIT (9), EXIN (1)).

M.E.T.R.O.-APEX File 513

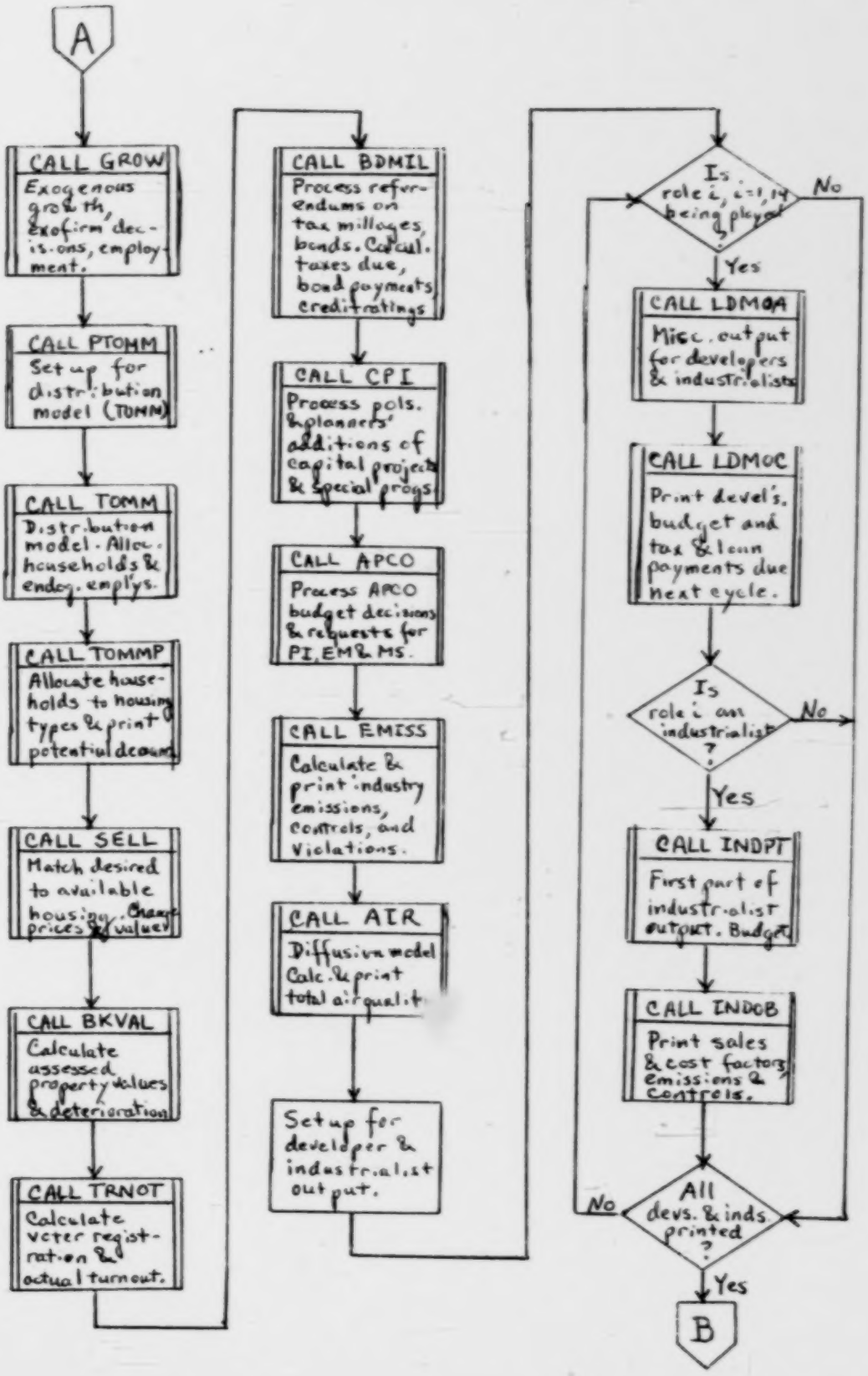
(Note: in the IBM 1130 version of METAPEX each record of this file was a third physical record in each logical record of file 1 only. It was made a separate file to facilitate the copy operation.) It is used only during the processing of the "current" cycle of the game but is retained in data set 15 until the beginning of the next cycle for possible error analysis.

Word	Type	Name	
1-154	I	OFRC (11, 14)	Property (units for 1-5 and 10ths of acres for 6-11) offered by the game realtor and industrialists in the AA in current cycle.
155-308	I	SORC (11,14)	Property (analogous to OFRC) actually sold by the game realtors and industrialists.
309-319	I	FOVER (11)	Corresponds to the 5 developed residential zoning categories, plus six endogenous commercial/industrial classifications. This array is used only in SELL and represents the unfilled demand for land (in 10ths of an acre) after all "cascading" or pre-empting of land by higher use types on the first four iterations.

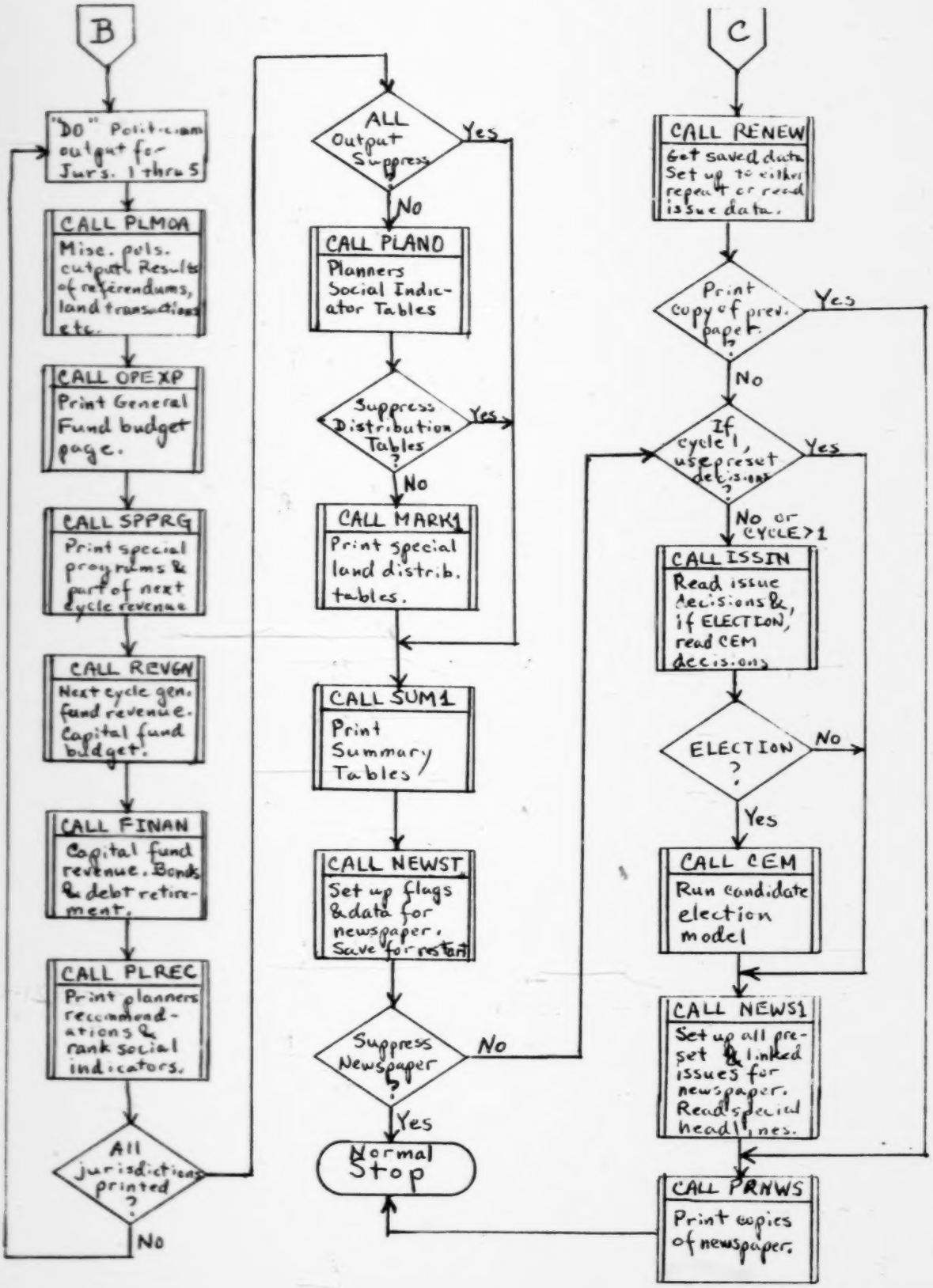
GENERAL PROGRAM FLOW
Main M.E.T.R.O.-APEX Program



General Flow Chart, Part 1


A
CALL GROW
Exogenous growth, exofirm decisions, employment.
CALL PTOMM
Set up for distribution model (TOMM)
CALL TOMM
Distribution model. Alloc. households & endog. empl'ys.
CALL TOMMP
Allocate households to housing types & print potential decouns
CALL SELL
Match desired to available housing. Change prices & values
CALL BKVAL
Calculate assessed property values & deterioration
CALL TRNOT
Calculate voter registration & actual turnout.
CALL BDMIL
Process referendums on tax milloges, bonds. Calcul. taxes due, bond payments, credit ratings
CALL CPI
Process pols. & planners' additions of capital projects & special progs.
CALL APCO
Process APCO budget decisions & requests for PI, EM & MS.
CALL EMISS
Calculate & print industry emissions, controls, and violations.
CALL AIR
Diffusion model Calc. & print total air qual.
Set up for developer & industrialist output.
Is role i, i=1,14 being played?
No
Yes
CALL LDMOA
Misc. output for developers & industrialists
CALL LDMOC
Print devel's. budget and tax & loan payments due next cycle.
Is role i an industrialist?
No
Yes
CALL INDPT
First part of industrialist output. Budget
CALL INDOB
Print sales & cost factors, emissions & controls.
All devs. & inds. printed?
No
Yes
B

General Flow Chart, Part 2



General Flow Chart, Part 3

MA/360
DFK-8/13/71

INSTRUCTIONS TO COMPUTER OPERATOR

Game Director _____ Computer Operator _____

Team Number _____, Cycle Number _____, Date Received _____

Instructions for Main Header Card:

__ NEWGAME. (Check only one of the four options and fill in number
__ CYCLE __, if appropriate.)
__ REPEAT __, (Underline cycle **0** if desired -- only preceding NEWGAME)
__ NEWSPAPER, [CYCLE 0,]

DATE __/__/__ (to be printed on output)

Team identification for page headings (required only if change from preceding cycle.)

'_____'

Program Control Card Options:

NOPLAY - DEV(1), __ IND(1), __
DEV(2), __ IND(2), __ (Check as appropriate. All are considered
DEV(3), __ IND(3), __ playing unless checked)
DEV(4), __ IND(5), __
DEV(5), __ IND(6), __
DEV(6), __ ALLIND, __
DEV(7), __
ALLDEV, __

SUPPRESS=APCO, __ DEV(1), __ DEV(2), __ DEV(3), __ DEV(4), __ DEV(5), __DEV(6), __
DEV(7), __

IND(1), __ IND(2), __ IND(3), __ IND(5), __ IND(6), __

POL(1), __ Central City Politicians
POL(5), __ County Politicians
PLANNER(1), __ Central City Planners
PLANNER(5), __ County Planners
DIST, __ Special Percent Distribution of Land Tables
SUMMARY, __ All Summary Tables
NEWS __
DATA, __ List of Data Input Cards (errors will be printed)

(All of the above will be printed unless suppressed.

=ALL, __ To suppress all of the above printing.

PRINT=POL(2), Suburb
POL(3), Township 1
POL(4), Township 2
PLANNER(2), Recommendations for Suburb
PLANNER(3), Recommendations for Township 1
PLANNER(4), Recommendations for Township 2
TOMMSTART, Diagnostics
TOMMEND, Diagnostics

(None of the above will be printed unless checked.)

DIFFUS=ALL
AUTUMN,
WINTER,
SPRING,
SUMMER,

(Printing of diffusion output for all AAs. Diagnostic output for operator. Suppressed unless checked.)

(The following options apply to newspaper and issue processing but must be included with main program control cards.)

ELECTION

CEM=DIAGNOSTICS, RESULTS, (Diagnostic results; does not effect printing in newspaper.)

BYPASS=CYCLEONE,
LINKS,
NATIONAL,
LOCAL,

Options for XX card following main player decision cards:

Normal -- stop if errors in data cards, go on if no errors (I-1 = blank)

Stop unconditionally after reading data cards (I-1= -1 negative)

Go on even if there are errors in data cards (I-1= 1 positive)

Newspaper Header Card:

Number of copies of newspaper desired: COPIES= ___,

IGNORE Ignore missing issue decisions

NOHEADS No special headline cards in input deck

REREAD Applicable only in conjunction with NEWSPAPER on Main Header Card. Read new issue (and, if ELECTION specified) candidate decision cards even though newspaper has already been printed. If REREAD NOT specified an exact copy will be printed.

Game Director or Operator additions to Main Player Decision Cards:

Change U.S. Employment Growth Rate Percentage:

Type Col.1-2 EG	Percent F-1

FORCE EXOFIRMS:

Code cols 1-2	Exofirm Number I-1	Force in to AA number* I-2
XF		
XF		
XF		
XF		
XF		
XF		

*If I-2 field blank, will force into exofirm's first choice of AAs.

ADDITIONAL EXOGENOUS EMPLOYMENT

Code Cols. 1-2	Analysis Area I-1	Type: 1=Manuf. 2=Bur I-2	Number of Employees F-1	No. of Acres F-2	Added Tax Base - $ F-3
EE					
EE					
EE					
EE					
EE					

ENTER BACKGROUND POLLUTION LEVELS:

Code Cols.	Season I-1	Particulated F-1	SO_2 F-2	CO F-3	NO_x F-4	HC F-5
BP	1=Winter					
BP	2=Spring					
BP	3=Summer					
BP	4=Fall					

OVERRIDE AUTOMATIC DIVIDEND CALCULATION:

Code	Industrialist Number I-1	Dollar amount of Dividends to be deducted F-1
DV		
DV		
DV		
DV		
DV		

CHANGE INDUSTRIALISTS" AVERAGE ANNUAL WAGE SCALE:

Place the amount of new annual wage in F-4 field of the PL card for the appropriate industrialist.

CHANGE STATE EQUALIZATION FACTOR (for assessment):

Code	Jurisdiction (1-4 only) I-1 Field	Assessment Proportion F-1 Field
AC		

CASH TRANSFERS TO OR FROM OPERATOR: Operator code is 99

Code	To whom I-1	For Whom I-2	Amount F-1
CT			
CT			
CT			
CT			

SPECIAL GRANTS:

Code	Jurisdiction I-1	Budget I-1	Amount F-1
SG			
SG			
SG			
SG			

Budget: 1 = General Operating
2 = Capital Fund

FEDERAL AIR POLLUTION CONTROL GRANT:

Code	Cols. 3-6	Current Cycle F-1	Next Cycle F-2	Cycle after next F-3
BR	0003			